Defense Acquisition University

Agile Project Management

Learn. Perform. Succeed.

Matthew R. Kennedy, PhD, CSP
Matthew.Kennedy@dau.mil

POTUS and SECDEF: "DoD Will Be Agile"

"The United States is going to maintain our military superiority with armed forces that are agile, flexible and ready for the full range of contingencies and threats."

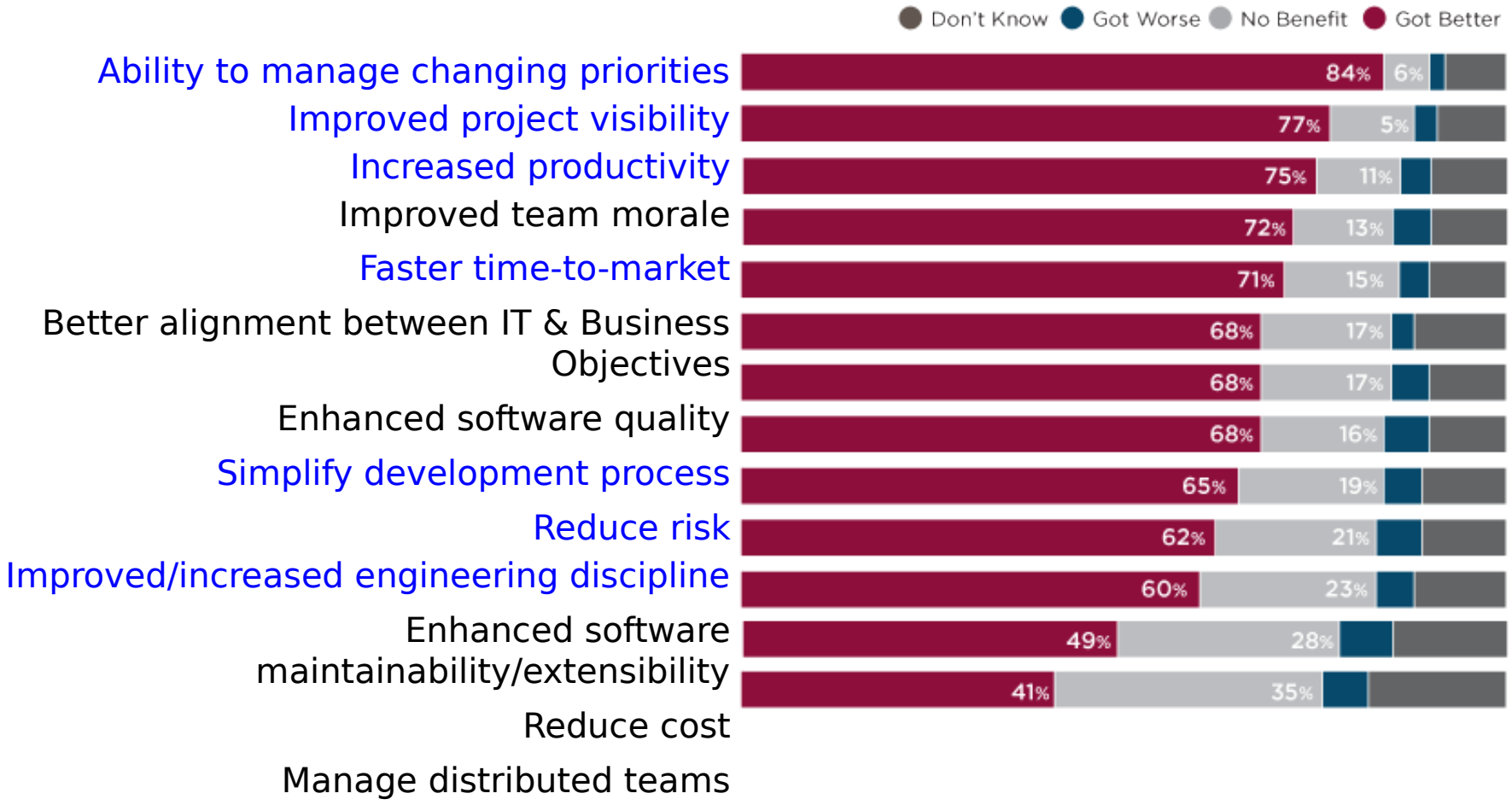
- President Obama

"The US joint force will be smaller and leaner. But its great strength will be that it will be more agile, more flexible, ready to deploy quickly, innovative, and technologically advanced. That is the force for the future."

- Secretary Panetta

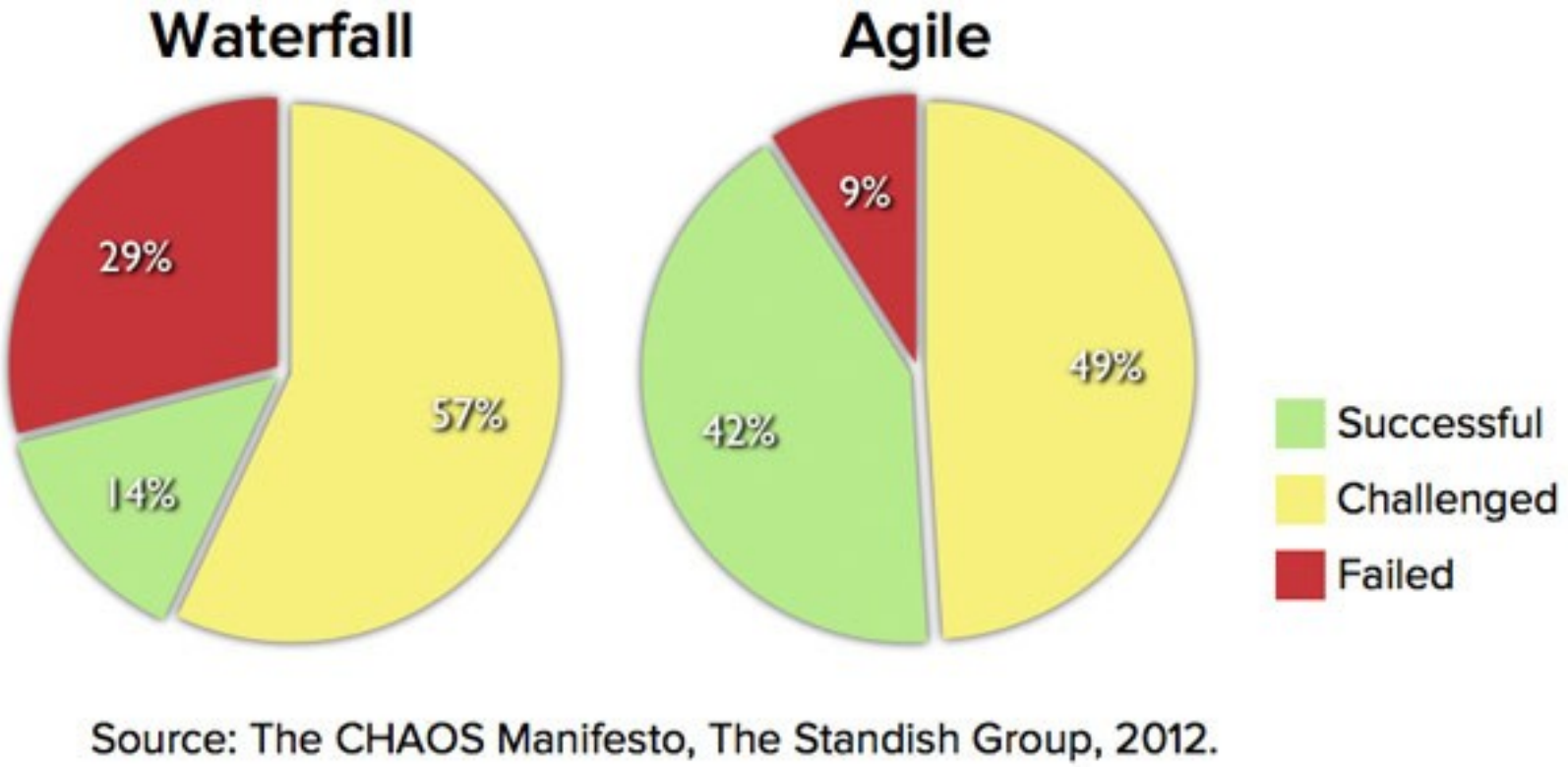
Defense Security Review, 5 Jan 12

Benefits from Implementing Agile



Source: Version One: State of Agile Survey 2011

Agile Does NOT Guarantee Success



Source: The CHAOS Manifesto, The Standish Group, 2012.

What is Agility?

- **“The speed of operations within an organization and speed in responding to customers (reduced cycle times)” (Mass. Inst. Tech.)**

Organizational Structure

Learn. Perform. Succeed.

Aspects of Product Development

- **Business Aspect**
  - **Responsible for the overall acquisition: contracting, funding, operational requirements, and system delivery structure**
- **Project / System Aspect**
  - **Overall technical management. Further decompose the requirements and allocate them to software or hardware**
- **Development Aspect**
  - **Developmental items**

Different Focus – Same Goal

Strategic Goals
Contracts
Funding
Op. Requirements

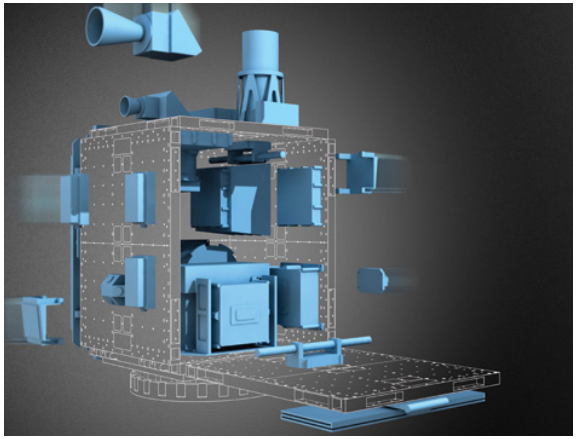
Tech. Requirements
Project Planning
Systems Planning
Technical Standards
Integration

Development Test

Business Aspect

Project / System Aspect

Development Aspect



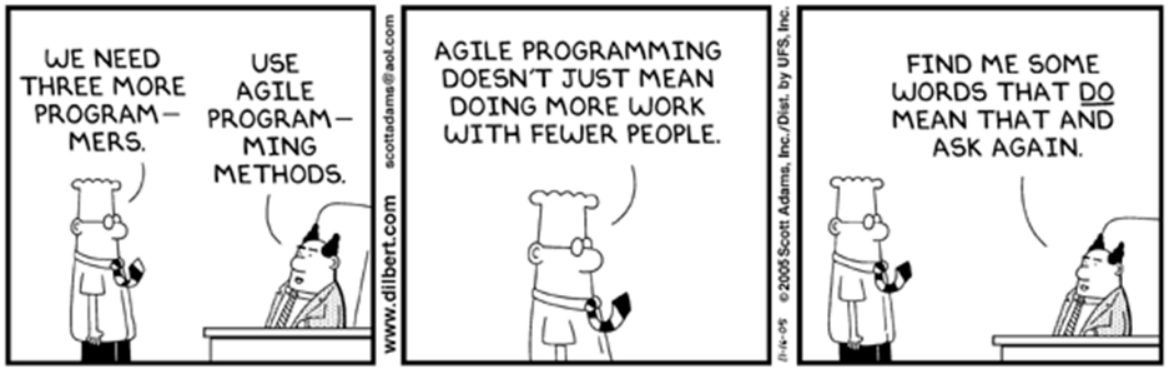
Projects are held captive by the slowest

Defense Acquisition University

Development Aspect

Agility within the Development Aspect

In the beginning…


WE NEED THREE MORE PROGRAM-MERS.
USE AGILE PROGRAM-MING METHODS.
www.dilbert.com scottadams@aol.com
AGILE PROGRAMMING DOESN'T JUST MEAN DOING MORE WORK WITH FEWER PEOPLE.
FIND ME SOME WORDS THAT DO MEAN THAT AND ASK AGAIN.
© 2005 Scott Adams, Inc./Dist. by UFS, Inc.

© Scott Adams, Inc./Dist. by UFS, Inc.

Agile Manifesto

Development Aspect

- **The foundational document for Agile software development**

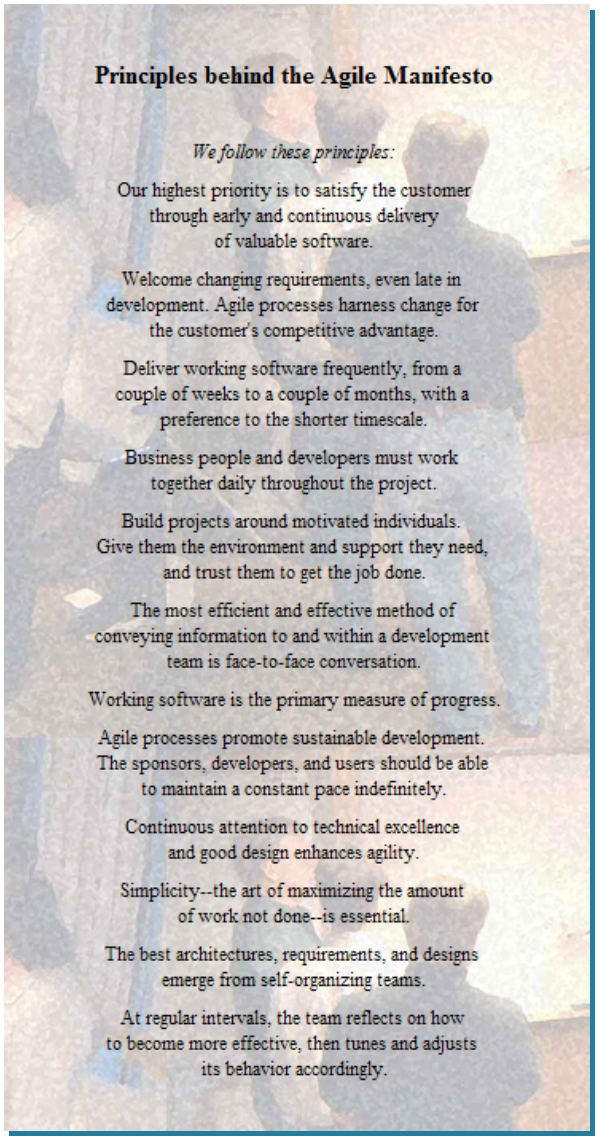
Manifesto for Agile Software Development
We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:
Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan
That is, while there is value in the items on the right, we value the items on the left more.
Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler
James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick
Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

- **Signed by 17 software developers in Feb 2001**
- **Core Values**
  - **Individuals and interactions** over processes and tools
  - **Working software** over comprehensive documentation
  - **Customer collaboration** over contract negotiation

http://agilemanifesto.org

12 Principles of the Agile Manifesto

Development Aspect


Principles behind the Agile Manifesto
We follow these principles:
Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
Business people and developers must work together daily throughout the project.
Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
Working software is the primary measure of progress.
Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
Continuous attention to technical excellence and good design enhances agility.
Simplicity--the art of maximizing the amount of work not done--is essential.
The best architectures, requirements, and designs emerge from self-organizing teams.
At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

1. **Continuous delivery of valuable software**
2. **Welcome changing requirements**
3. **Deliver working software in weeks/months**
4. **Work together daily**
5. **Build projects around motivated individuals**
6. **Face-to-face conversation**
7. **Working software is the measure of progress**
8. **Promote sustainable development**
9. **Good design enhances agility**

http://agilemanifesto.org

Agile Software Development Methodologies

Development Aspect

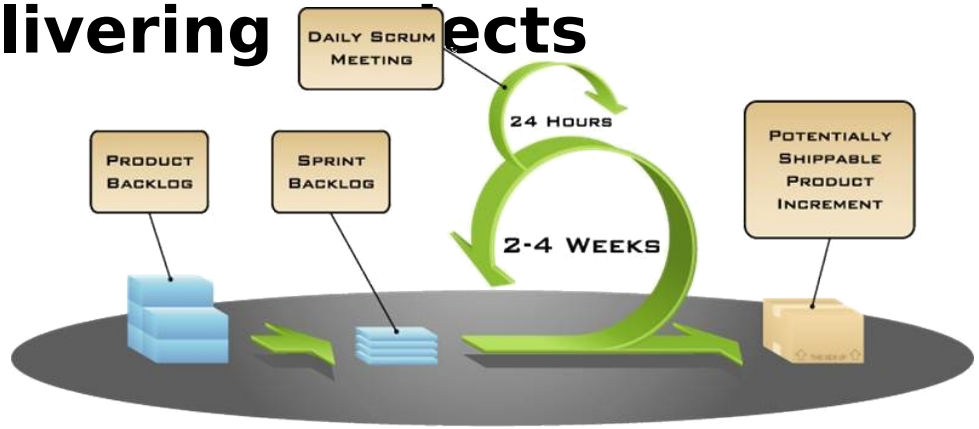
- Scrum
- eXtreme Programming (XP)
- Dynamic Systems Development Method
- Rapid Application Development
- Crystal
- Kanban
- …

WE'RE GOING TO TRY SOMETHING CALLED AGILE PROGRAMMING.
THAT MEANS NO MORE PLANNING AND NO MORE DOCUMENTATION. JUST START WRITING CODE AND COMPLAINING.
I'M GLAD IT HAS A NAME.
THAT WAS YOUR TRAINING.
www.dilbert.com scottadams@aol.com
© 2007 Scott Adams, Inc./Dist. by UFS, Inc.

© Scott Adams, Inc./Dist. by UFS, Inc.

Scrum

- **Project management focused**
- **One of the “agile processes”**
- **Self-organizing teams**
- **Product progresses in a series of “sprints”**
- **Requirements are captured as items in a list of “product backlog”**
- **No specific engineering practices prescribed**
- **Uses generative rules to create an agile environment for delivering projects**



Copyright © 2005, Mountain Goat Software

Scrum – Product Backlog

Development Aspect

- A user **Prioritized list** of requirements
- Time estimates completed by the te[illegible]he work

	Item #	Description	Est	By
Very High				
	1	**Finish database versioning**	16	KH
	2	**Get rid of unneeded shared Java in database**	8	KH
	-	**Add licensing**	-	-
	3	Concurrent user licensing	16	TG
	4	Demo / Eval licensing	16	TG
		Analysis Manager		
	5	File formats we support are out of date	160	TG
	6	Round-trip Analyses	250	MC
High				
	-	**Enforce unique names**	-	-
	7	In main application	24	KH
	8	In import	24	AM
	-	**Admin Program**	-	-
	9	Delete users	4	JM
	-	**Analysis Manager**	-	-
	10	When items are removed from an analysis, they should show up again in the pick list in lower 1/2 of the analysis tab	8	TG
	-	**Query**	-	-
	11	Support for wildcards when searching	16	T&A
	12	Sorting of number attributes to handle negative numbers	16	T&A
	13	Horizontal scrolling	12	T&A
	-	**Population Genetics**	-	-
	14	Frequency Manager	400	T&M
	15	Query Tool	400	T&M
	16	Additional Editors (which ones)	240	T&M
	17	Study Variable Manager	240	T&M
	18	Haplotypes	320	T&M
	19	**Add icons for v1.1 or 2.0**	-	-
	-	**Pedigree Manager**	-	-
	20	Validate Derived kindred	4	KH
Medium				
	-	**Explorer**	-	-
	21	Launch tab synchronization (only show queries/analyses for logged in users)	8	T&A
	22	Delete settings (?)	4	T&A

Copyright © 2005, Mountain Goat Software

Scrum – Sprint Backlog

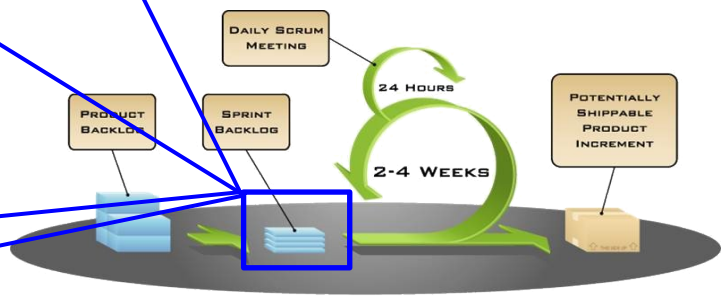
Development Aspect

- **Subset of items from the product backlog**
- **Based on teams velocity**

Sprint Backlog

	Item #	Description	Est	By
Very High				
	1	Finish database versioning	16	KH
	2	Get rid of unneeded shared Java in database	8	KH
	-	Add licensing	-	-
	3	Concurrent user licensing	16	TG
	4	Demo / Eval licensing	16	TG
		Analysis Manager		
	5	File formats we support are out of date	160	TG
	6	Round-trip Analyses	250	MC
High				
	-	Enforce unique names	-	-
	7	In main application	24	KH
	8	In import	24	AM
	-	Admin Program	-	-
	9	Delete users	4	JM
	-	Analysis Manager	-	-
	10	When items are removed from an analysis, they should show up again in the pick list in lower 1/2 of the analysis tab	8	TG
	-	Query	-	-
	11	Support for wildcards when searching	16	T&A
	12	Sorting of number attributes to handle negative numbers	16	T&A
	13	Horizontal scrolling	12	T&A
	-	Population Genetics	-	-
	14	Frequency Manager	400	T&M
	15	Query Tool	400	T&M
	16	Additional Editors (which ones)	240	T&M
	17	Study Variable Manager	240	T&M
	18	Haplotypes	320	T&M
	19	Add icons for v1.1 or 2.0	-	-
	-	Pedigree Manager	-	-
	20	Validate Derived kindred	4	KH
Medium				
	-	Explorer	-	-
	21	Launch tab synchronization (only show queries/analyses for logged in users)	8	T&A
	22	Delete settings (?)	4	T&A

If the teams Velocity was: 275

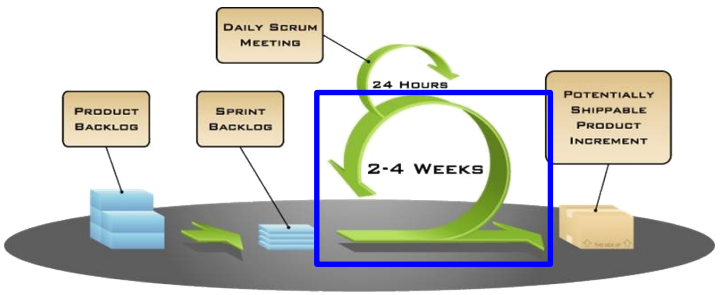


Copyright © 2005, Mountain Goat Software

Scrum – The Sprint

Development Aspect

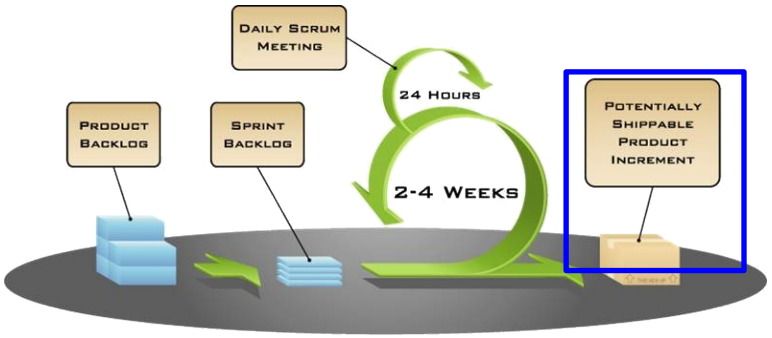
- **Requirements CAN NOT change during the Sprint**
- **Reflect daily on current status**
  - **What did you do yesterday?**
  - **What are you going to do tomorrow?**
  - **Did anything get in your way?**



Copyright © 2005, Mountain Goat Software

Scrum – Product

- **Potentially shippable product:**
  - **“Definition of Done”**
    - **Documented**
    - **Tested**
    - **Passed Information Assurance (IA)**
    - **Etc…**
- **Accepted by the user**
  - **If not accepted or “Done” - the item goes back into the product backlog and NO credit is given**



Copyright © 2005, Mountain Goat Software

Business aspect

Learn. Perform. Succeed.

Fast, Inexpensive, Simple, Tiny (F.I.S.T.) Manifesto*

- **Similar to the Agile Manifesto but targets the Business Aspect**
- **Values**
  - **Talent trumps process**
  - **Teamwork trumps paperwork**
  - **Leadership trumps management**
  - **Trust trumps oversight**

* Lt. Col. Dan Ward: DAU Publications

F.I.S.T.

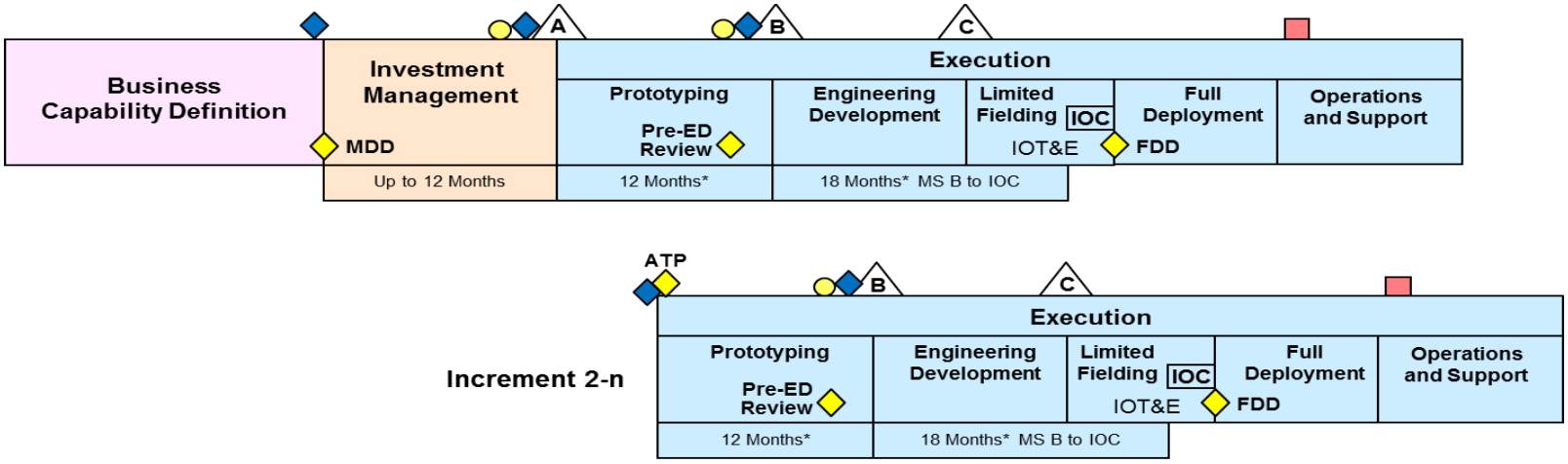
Business Aspect.

- **Principles**
  - **A project leader's influence is inversely proportional to the project's budget and schedule;**
  - **Constraints foster creativity;**
  - **Fixed funding and floating requirements are better than fixed requirements and floating funding;**
  - **An optimal failure costs a little and teaches a lot;**
  - **Complexity is cost, Complexity reduces reliability, Simplicity scales Complexity does not;**
  - **Iteration drives learning, discovery and efficiency**
- **Implementation Guidelines**
  - **Minimize team size and maximize team talent;**
  - **Use schedules and budgets to constrain the desig**
  - **Insist on simplicity in organizations, processes ar**
  - **Incentivize and reward under runs;**
  - **Requirements must be achievable within short time horizons;**
  - **Designs must only include mature technologies;**
  - **Documents and meetings: have as many as necessary, as few as possible;**
  - **Delivering useful capabilities is the only measure of Success**

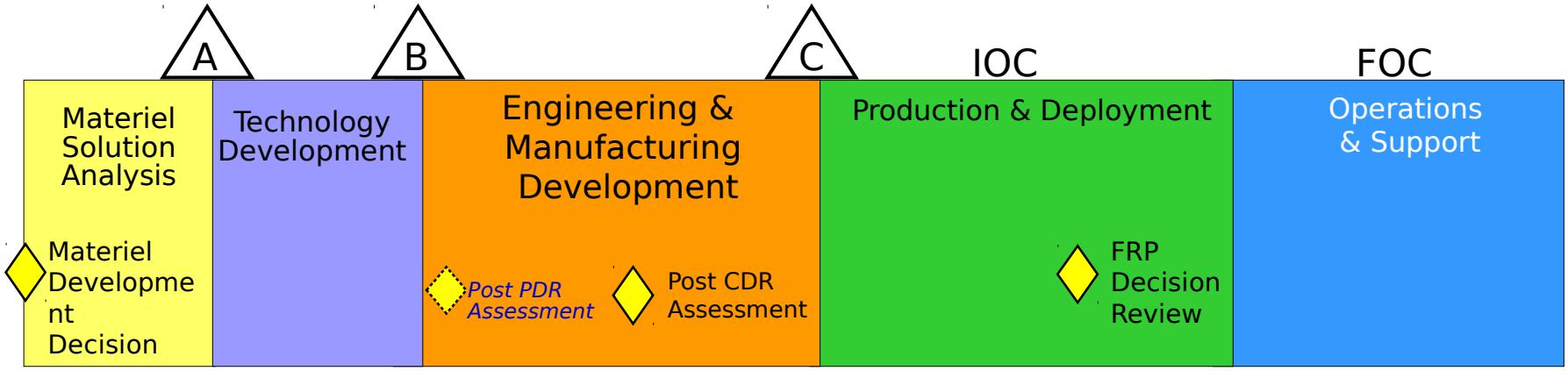
Fixed Funding + Fixed Requirements + Fixed Schedule by Definition IS NOT AGIL

Business Aspect Frameworks

- **Business Capabilities Lifecycle Framework**



- **DoD 5000.02**



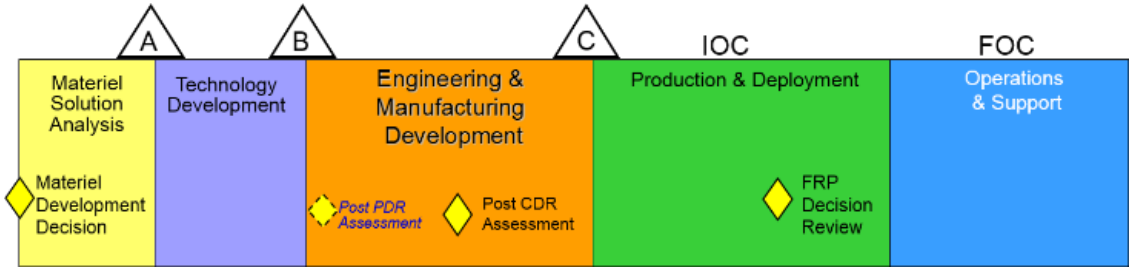
Project / System Aspect

Learn. Perform. Succeed.

How do we…

Project / System Aspect

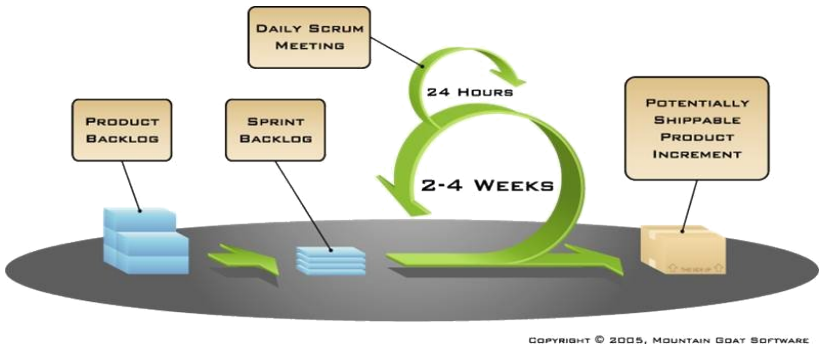
Business Aspect



Project / System Aspect

?

Software Aspect



Agile Practices

Incremental Development	**Small Teams**
Iterative Development	**Time Boxing**
Short Time-lines	**Lean Initiatives**
Retrospectives (Lessons learned)	**Prototyping**
Empowered / Self-organizing / Managing teams	**Continuous User Involvement**
Prioritized Product Backlog (Requirements)	**Co-located Teams**

Kennedy / Ward

"Traditional" Project Plan

Project / System Aspect

Business Aspect

A

B

C

IOC

FOC

Materiel Solution Analysis

Technology Development

Engineering & Manufacturing Development

Production & Deployment

Operations & Support

Materiel Development Decision

Post PDR Assessment

Post CDR Assessment

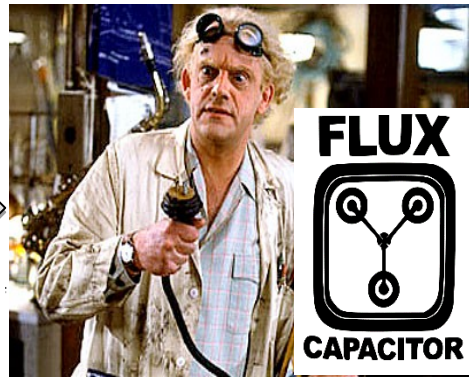
FRP Decision Review

Project / System Aspect

Provides

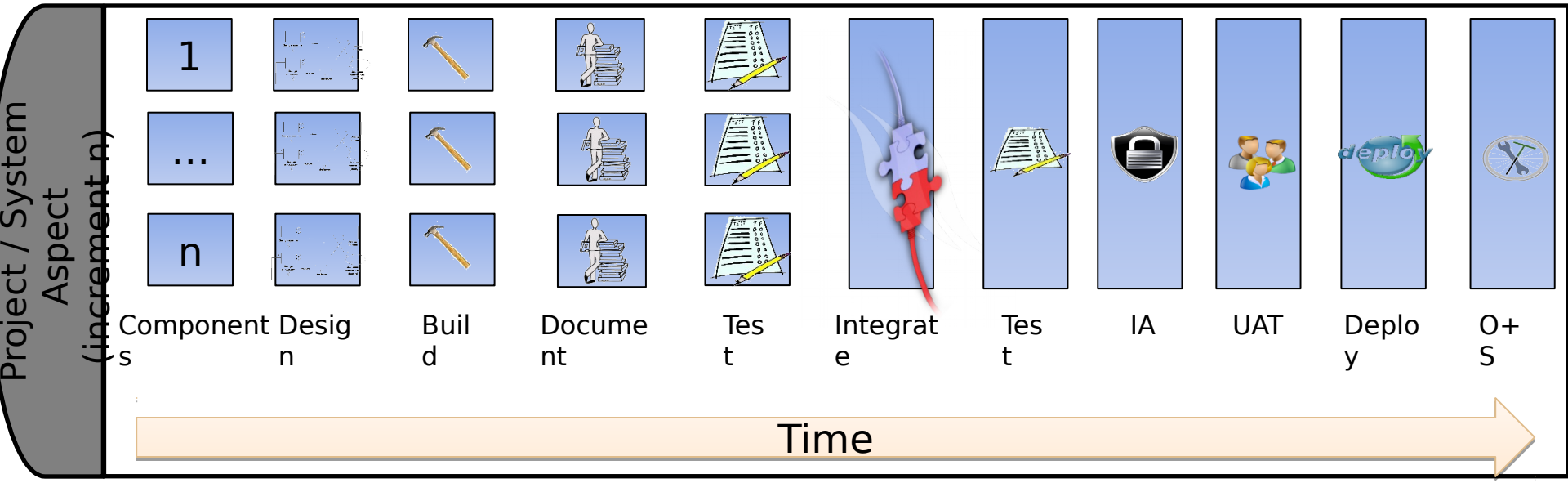
Stable Funding
Flexible Contracts
Flexible Requirements
Incremental Deliveries
Etc…

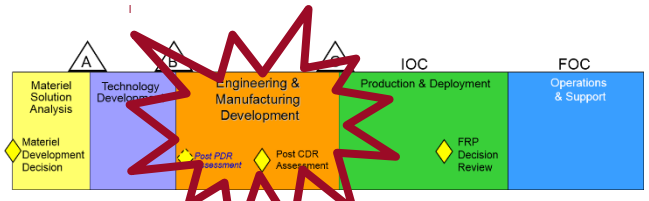
Enables



“Traditional” Project Planning Meeting

Project / System Aspect

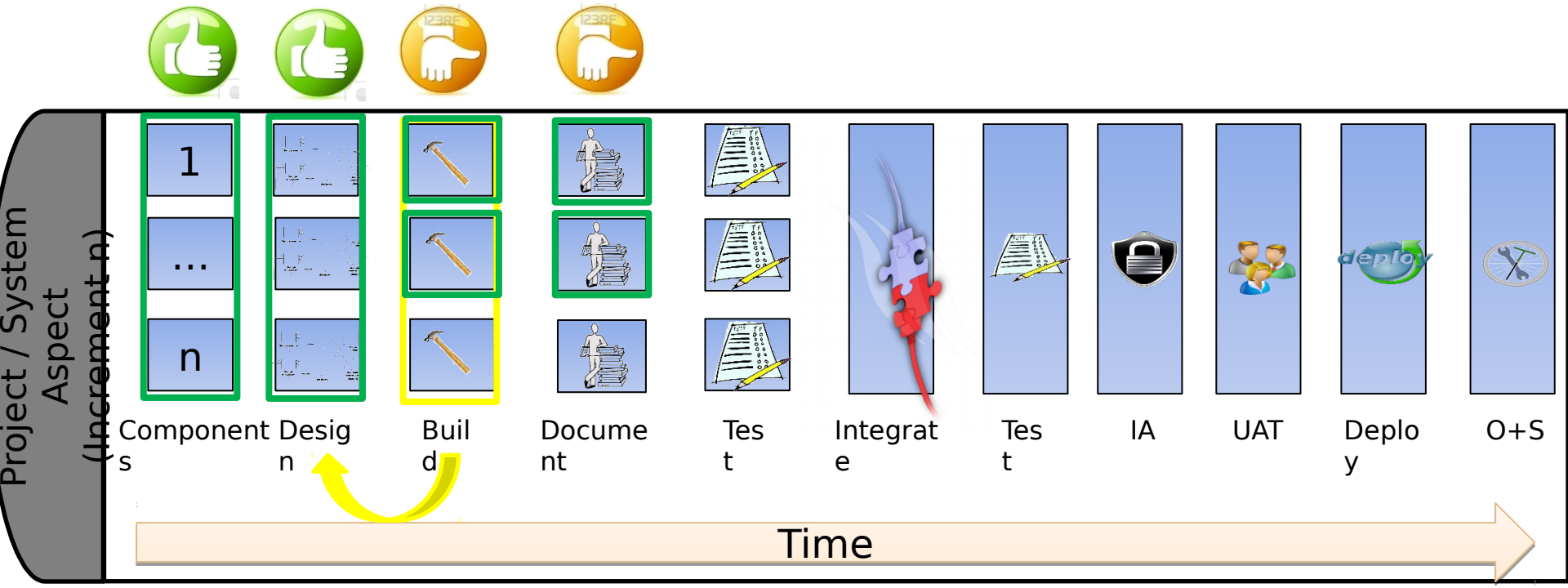


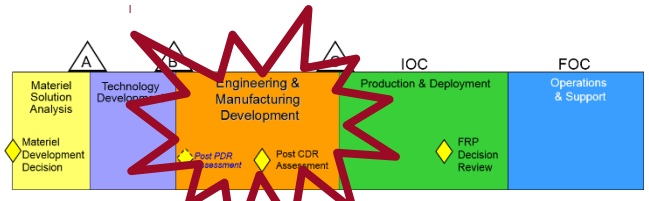


Success.

“Traditional” Project Execution

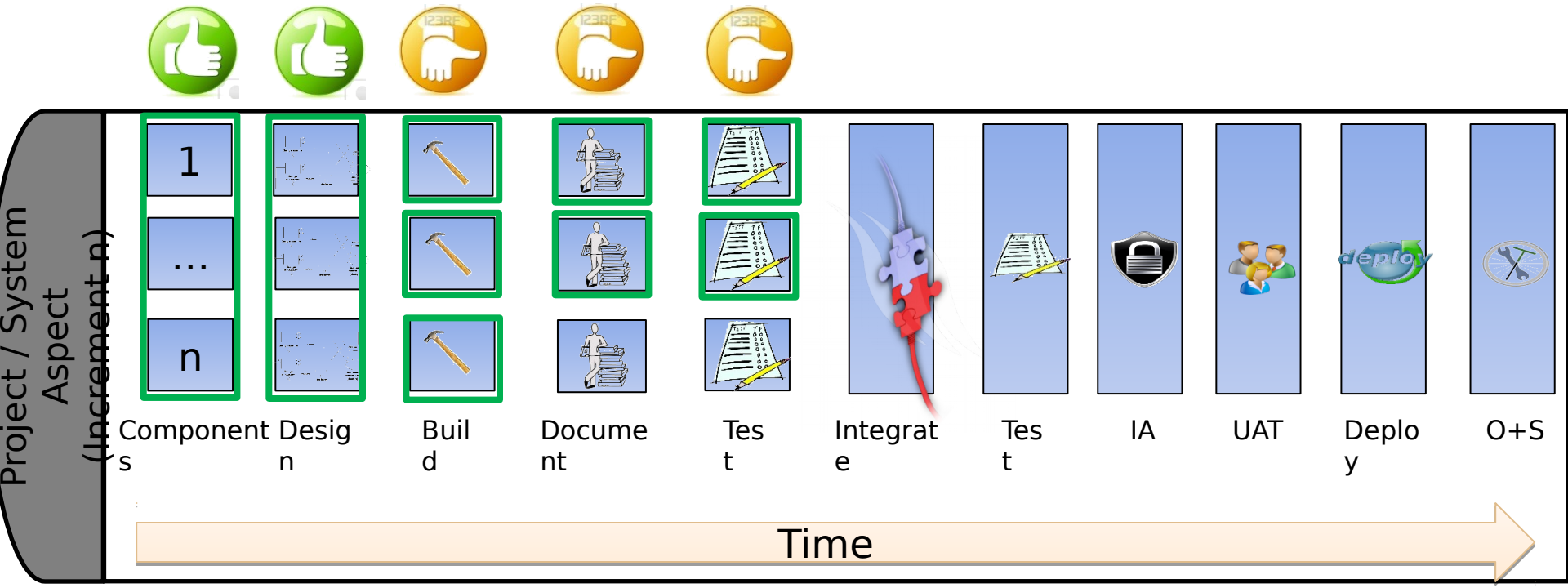
Project / System Aspect

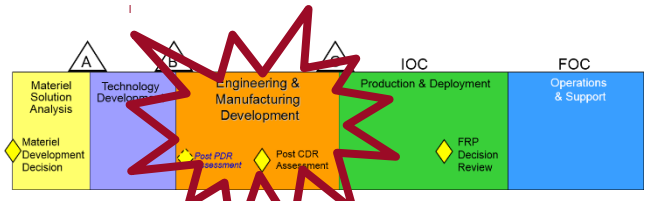




“Traditional” Project Execution

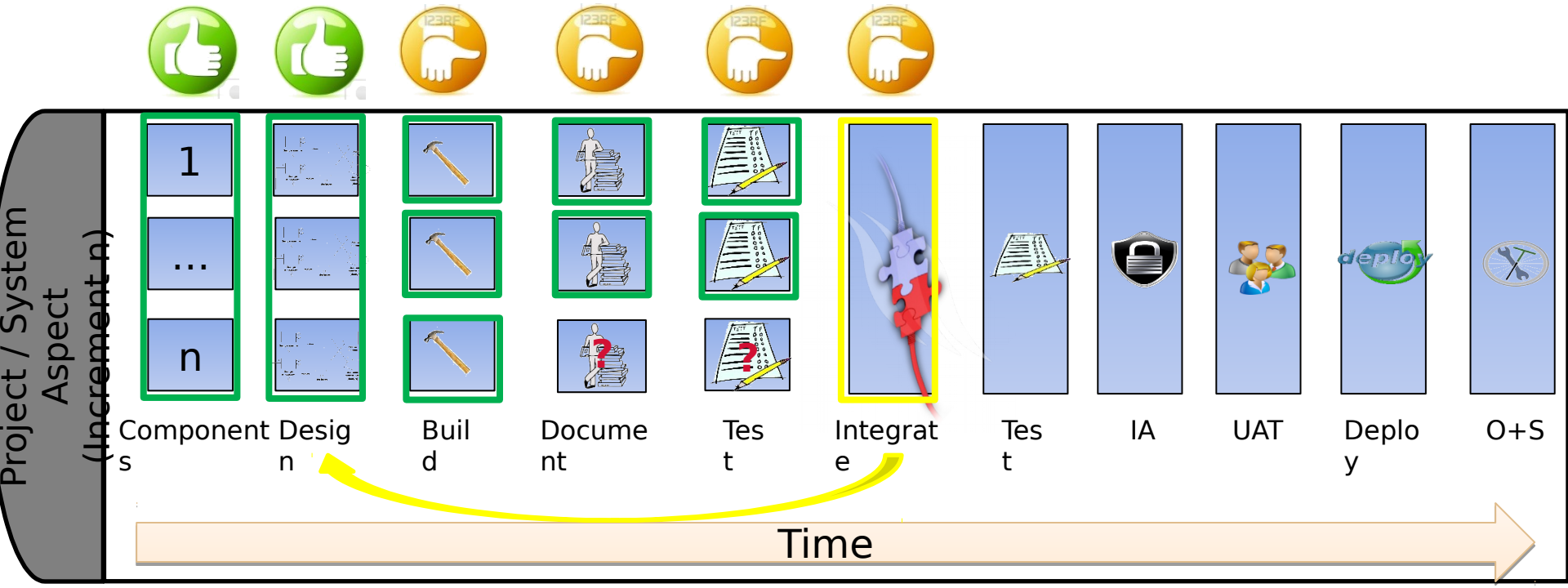
Project / System Aspect

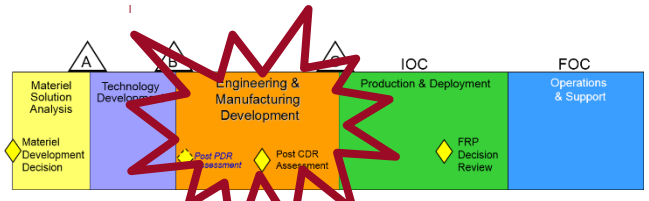




“Traditional” Project Execution

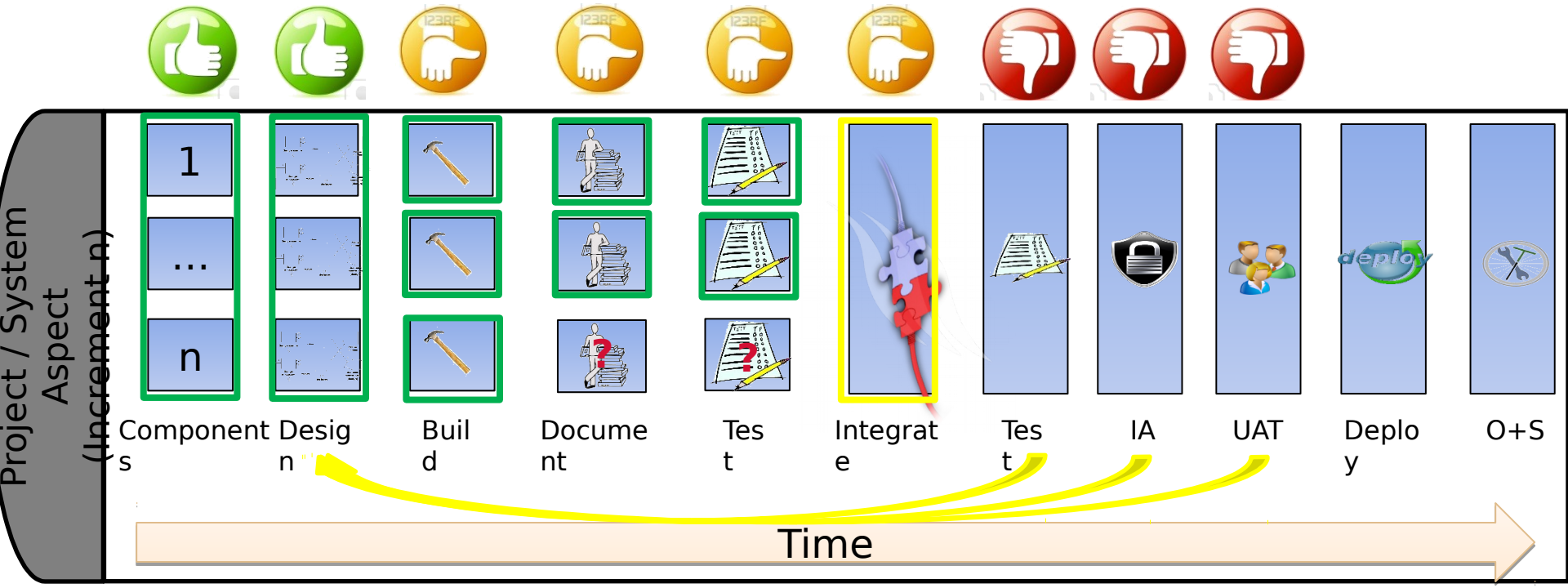
Project / System Aspect

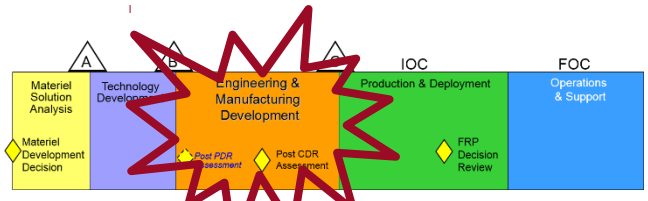




“Traditional” Project Execution

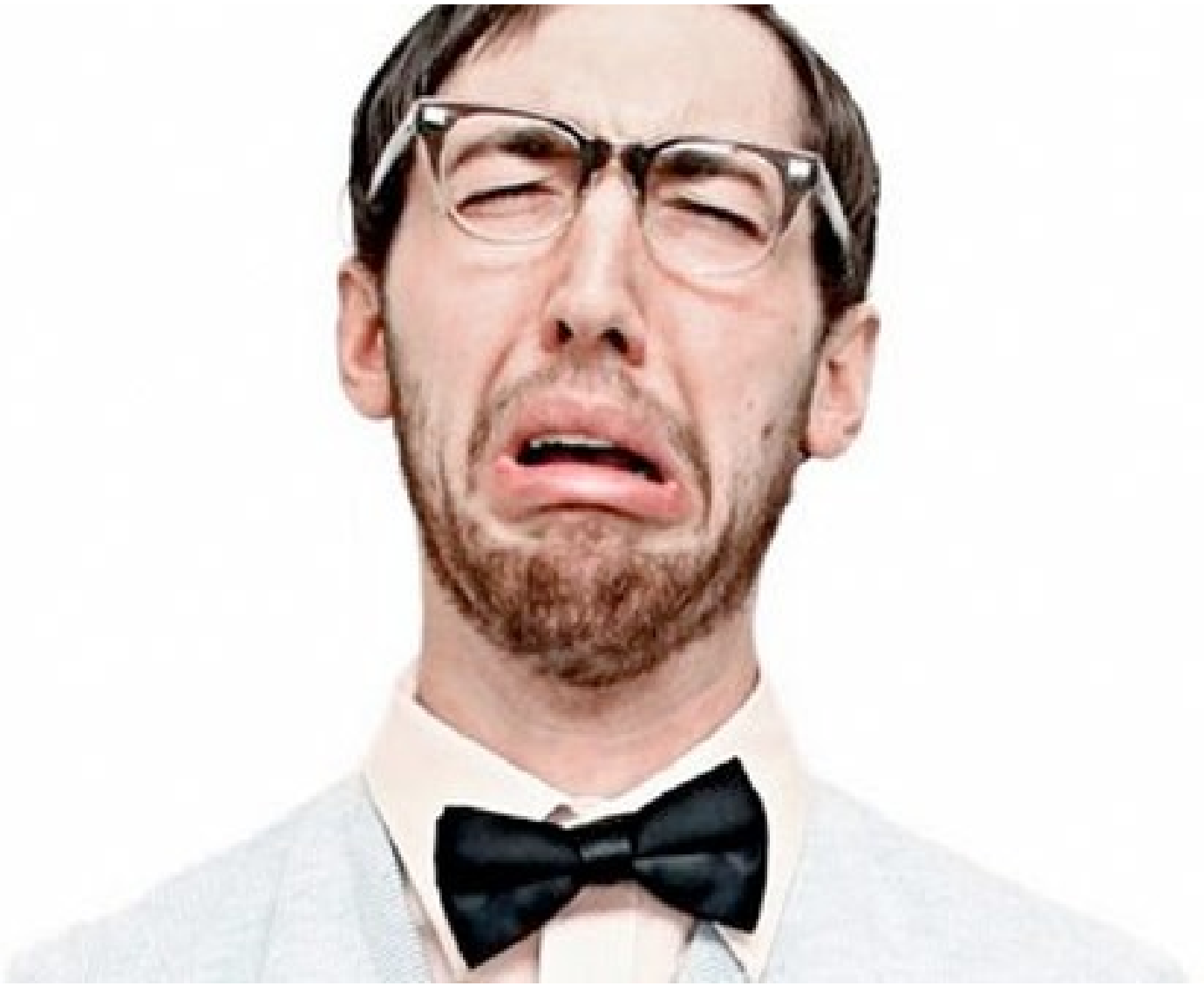
Project / System Aspect





RESULTS!

Project / System Aspect



“Programmed” Thought Process

Project / System Aspect

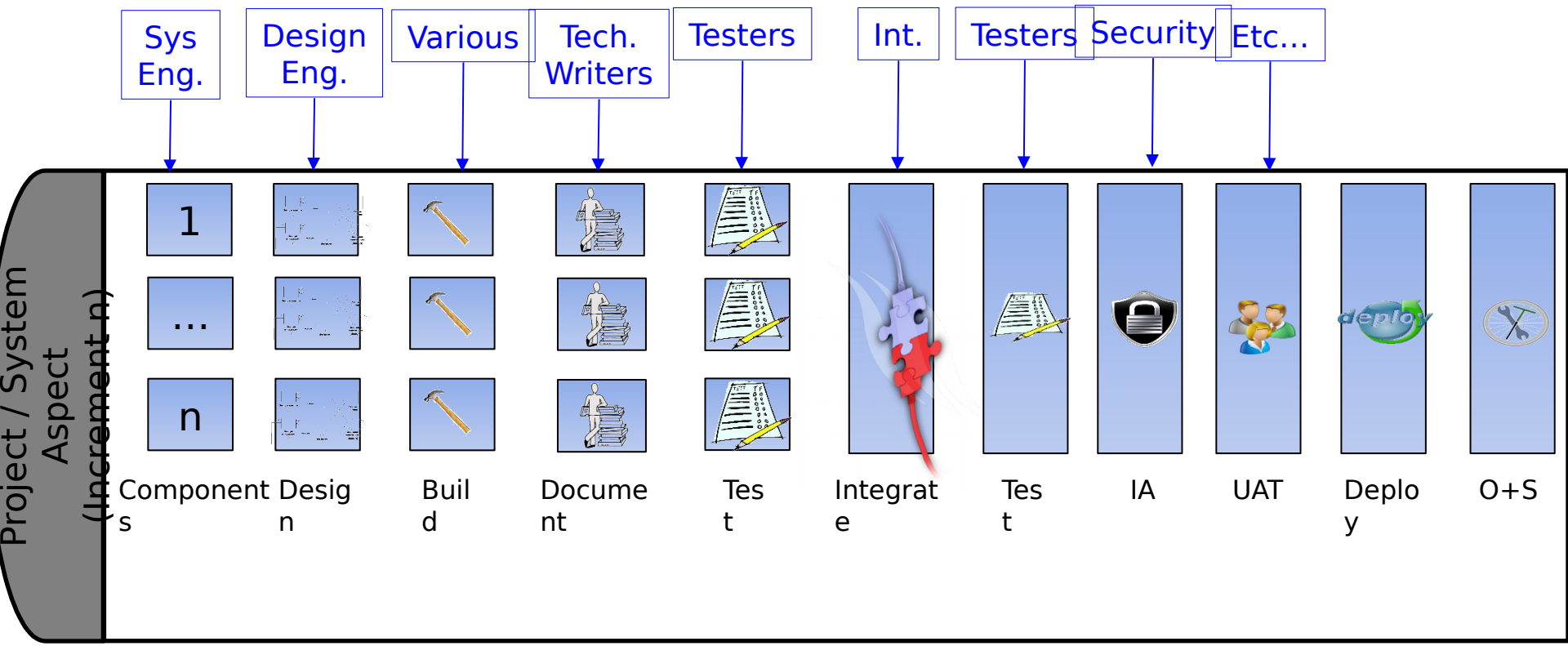
Step 1: Go to College

Project / System Aspect

Step 2: Get a Job

Project / System Aspect

Step 3: Develop a Product



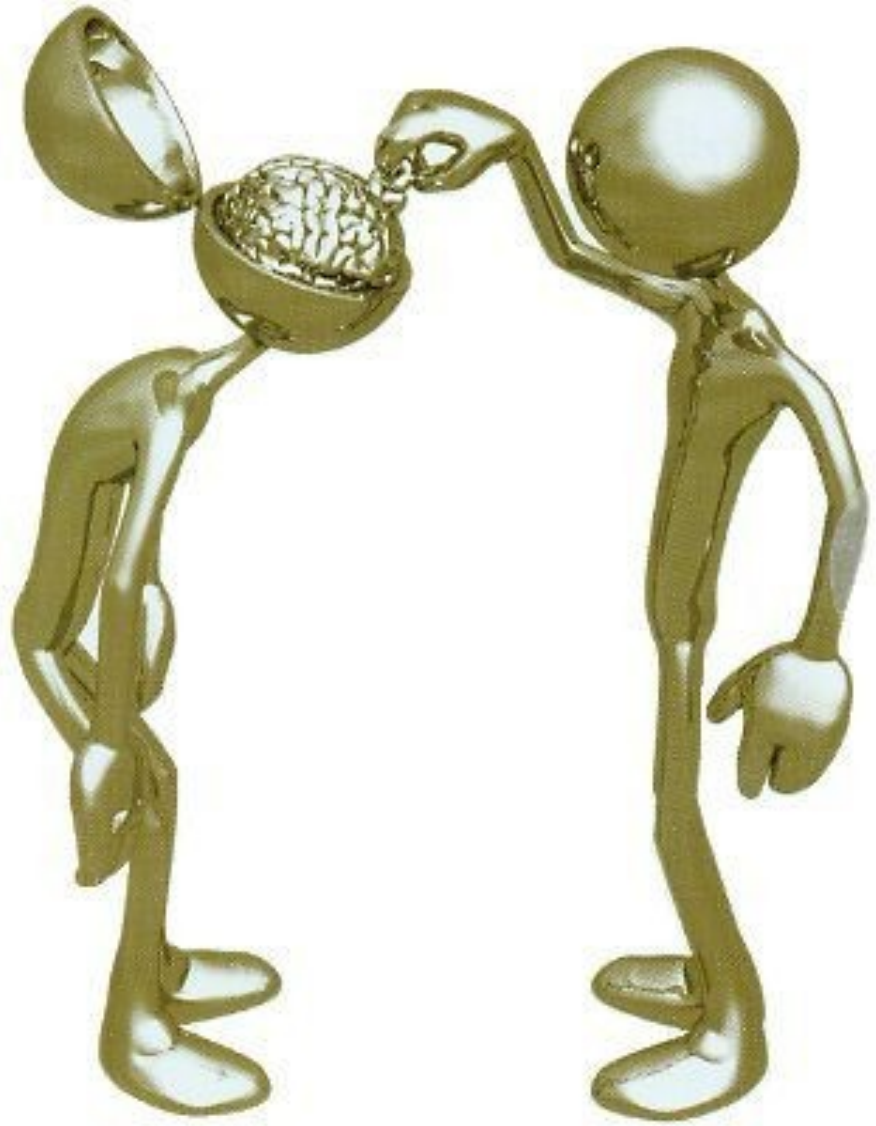
Everything is Stovepiped

Project / System Aspect

- **Even Our Contracts**
  - **Development Contract**
  - **QA Contract**
  - **Technical Writing Contract**
  - **Etc…**

Rewire How We Think

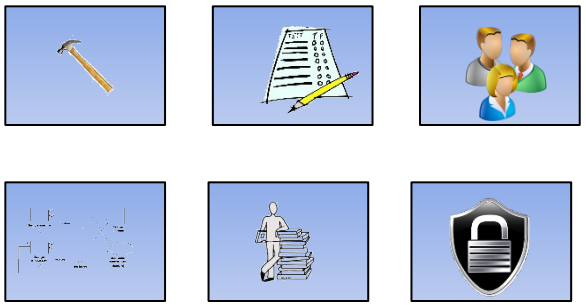
Project / System Aspect



Agile Uses Multidisciplinary Teams

Project / System Aspect

Traditional



Domain Focused

Agile

Capability Focused

The 'New' Cyber Security Domain

Project / System Aspect

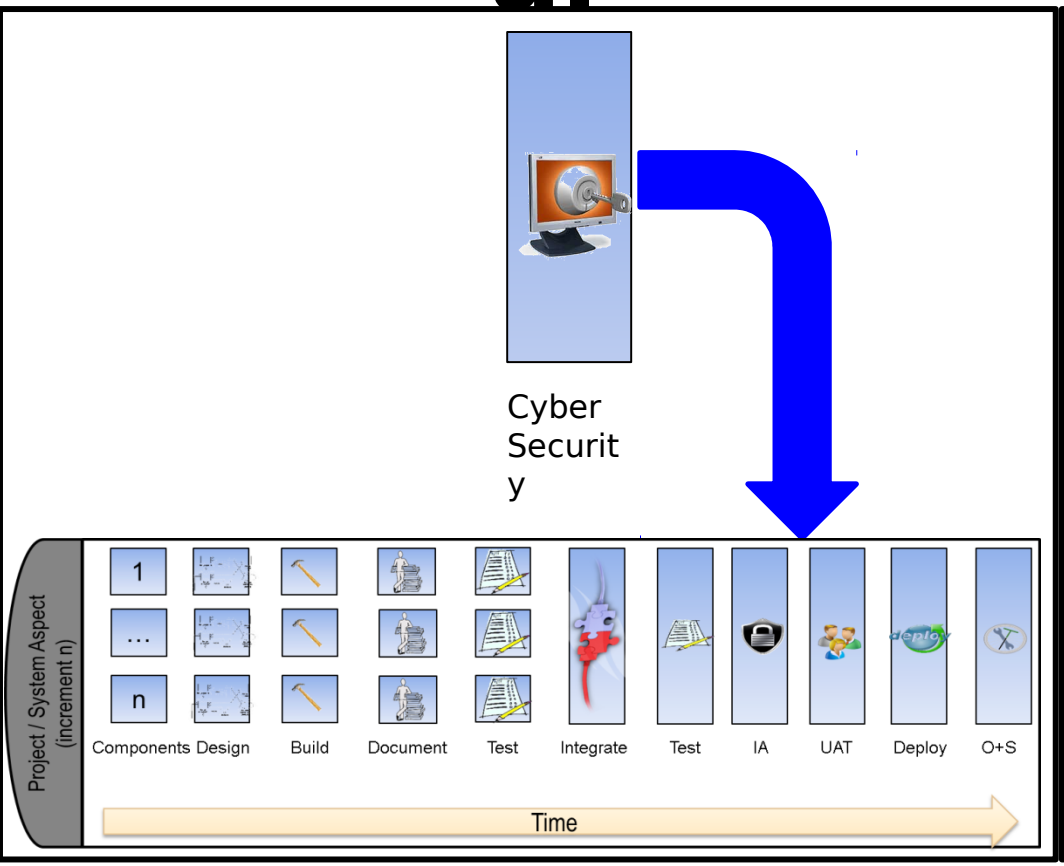
- **Universities are offering degrees in Cyber Security**
- **Federal Government is standing up an FFRDC for Cyber Security**
- **Departments have cyber security centers of excellence**

How will we implement this 'new' Cyber Security Domain?

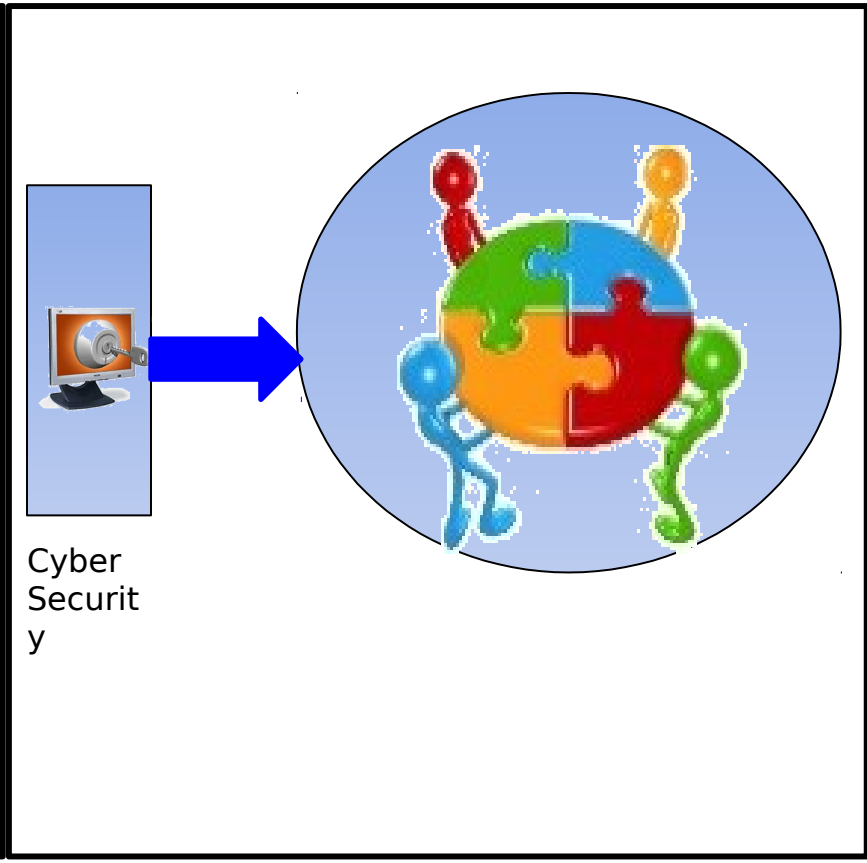
How will we respond?

Project / System Aspect

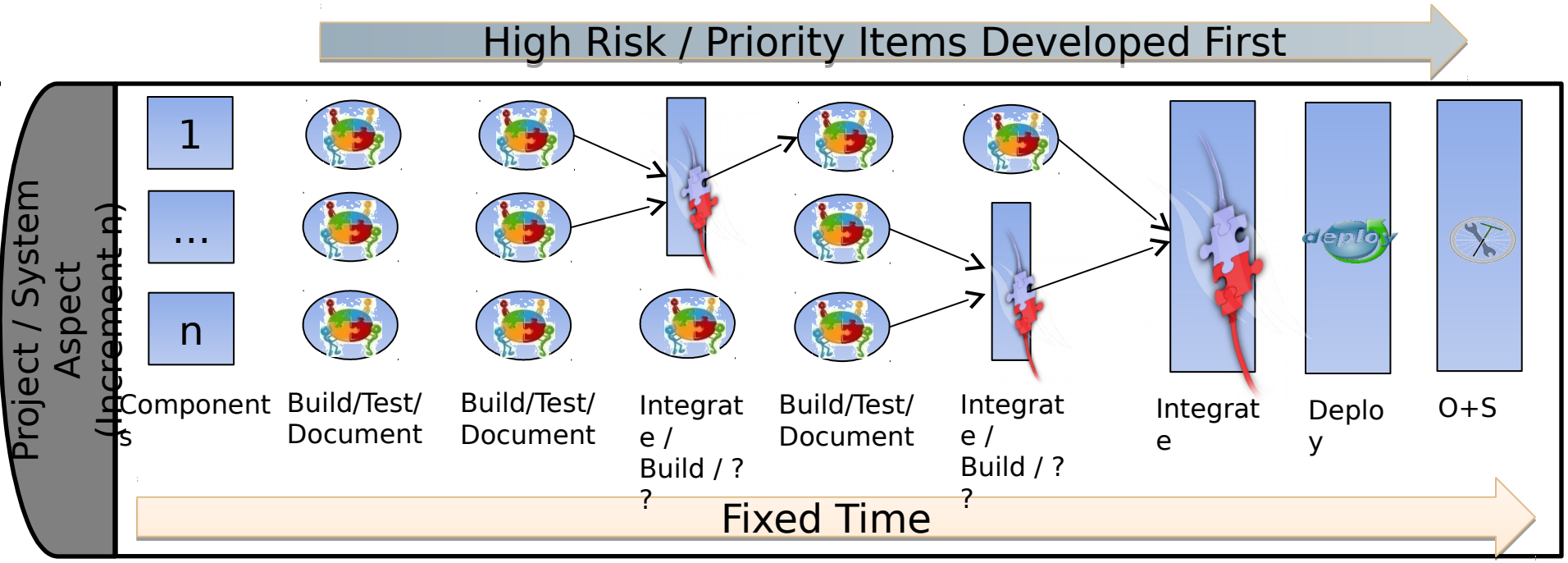
Traditional



Agile



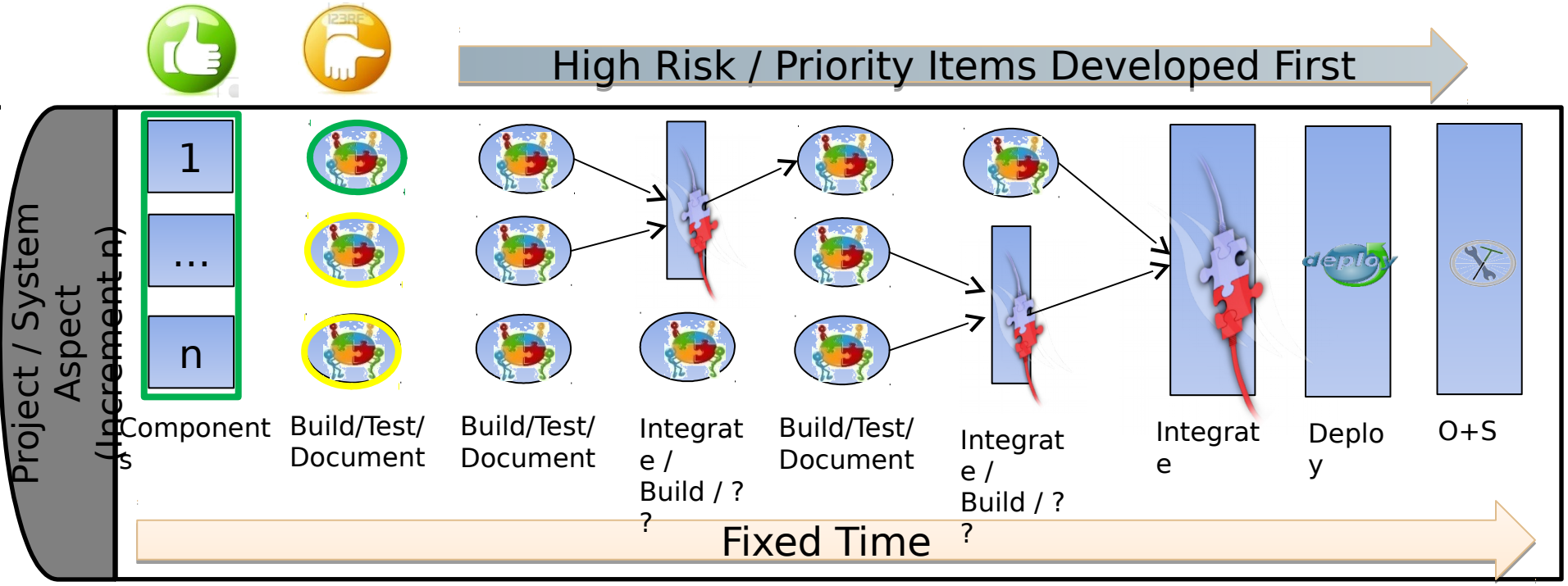
“Agile” Project Planning Meeting



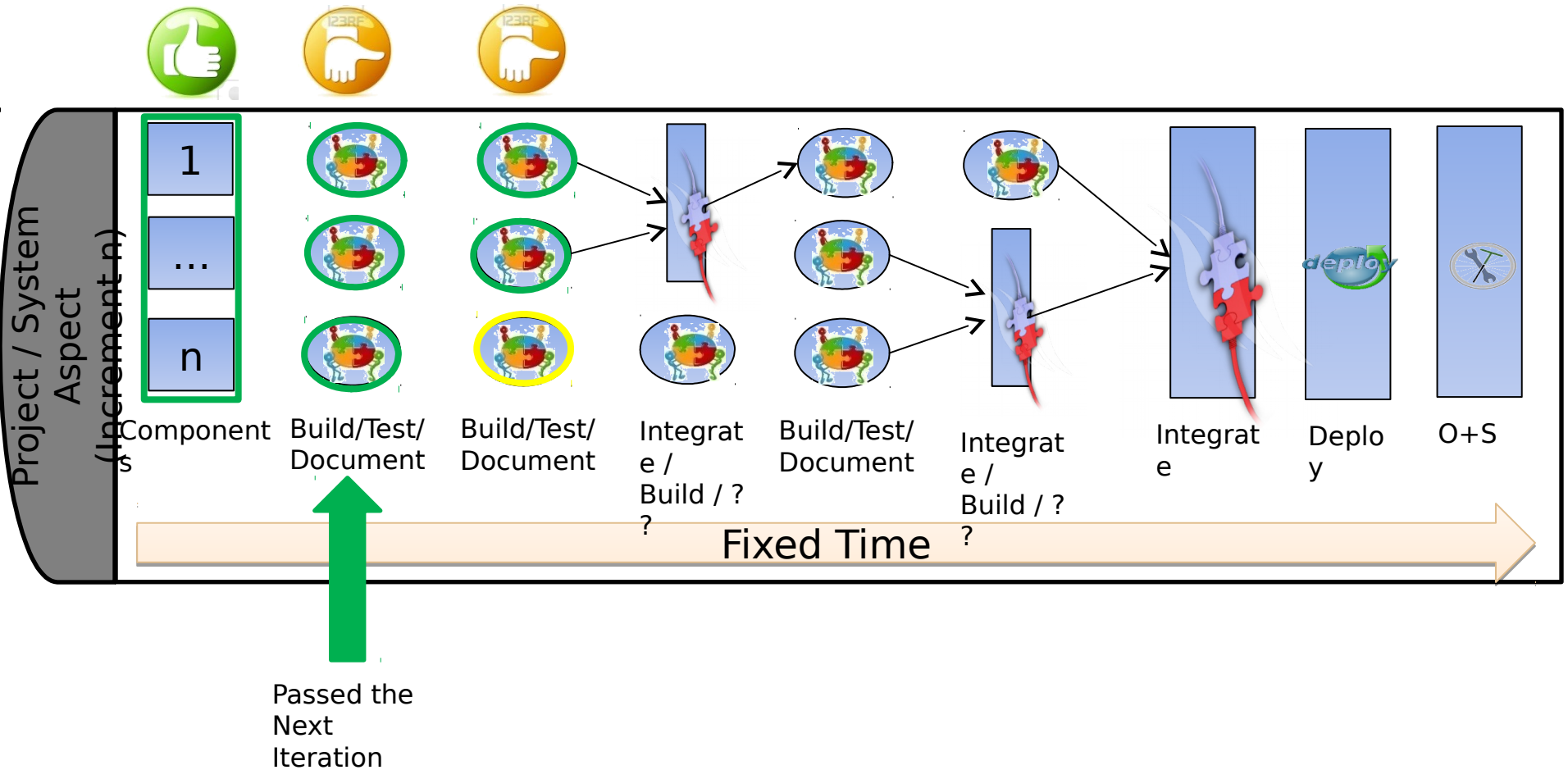
Success.

“Agile” Project Execution

Project / System Aspect

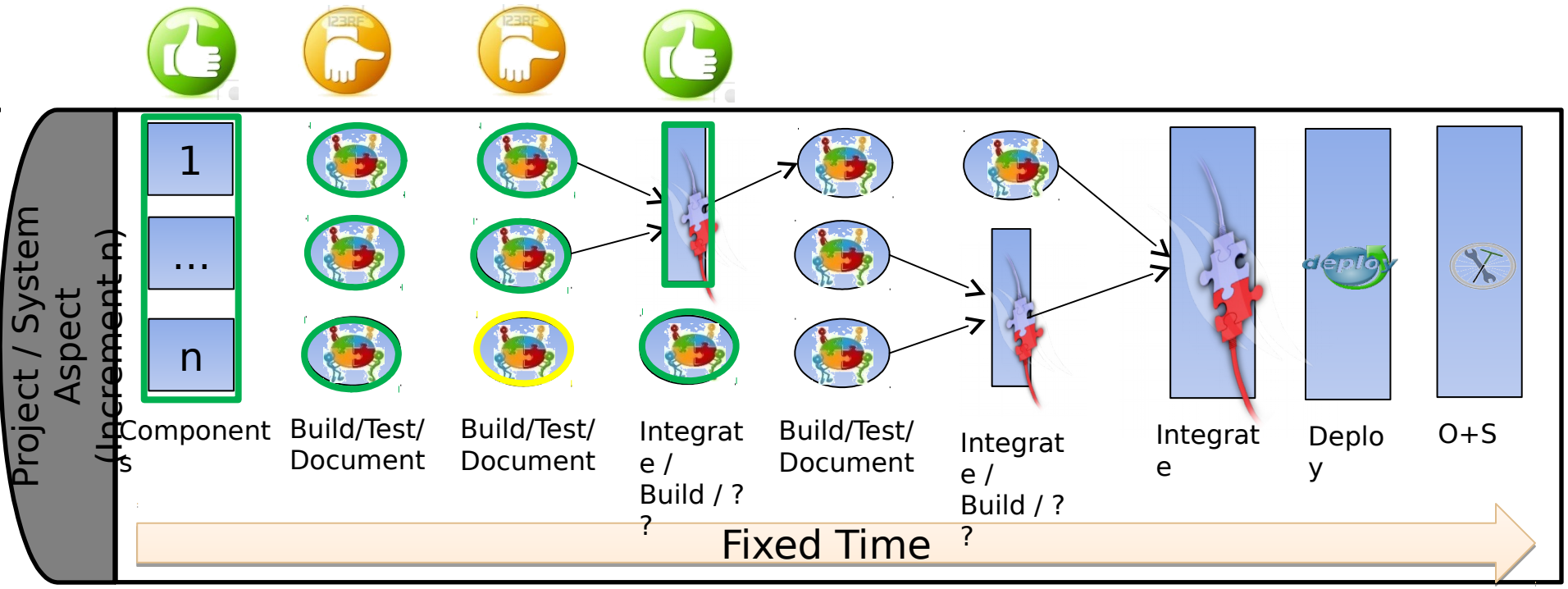


“Agile” Project Execution



“Agile” Project Execution

Project / System Aspect



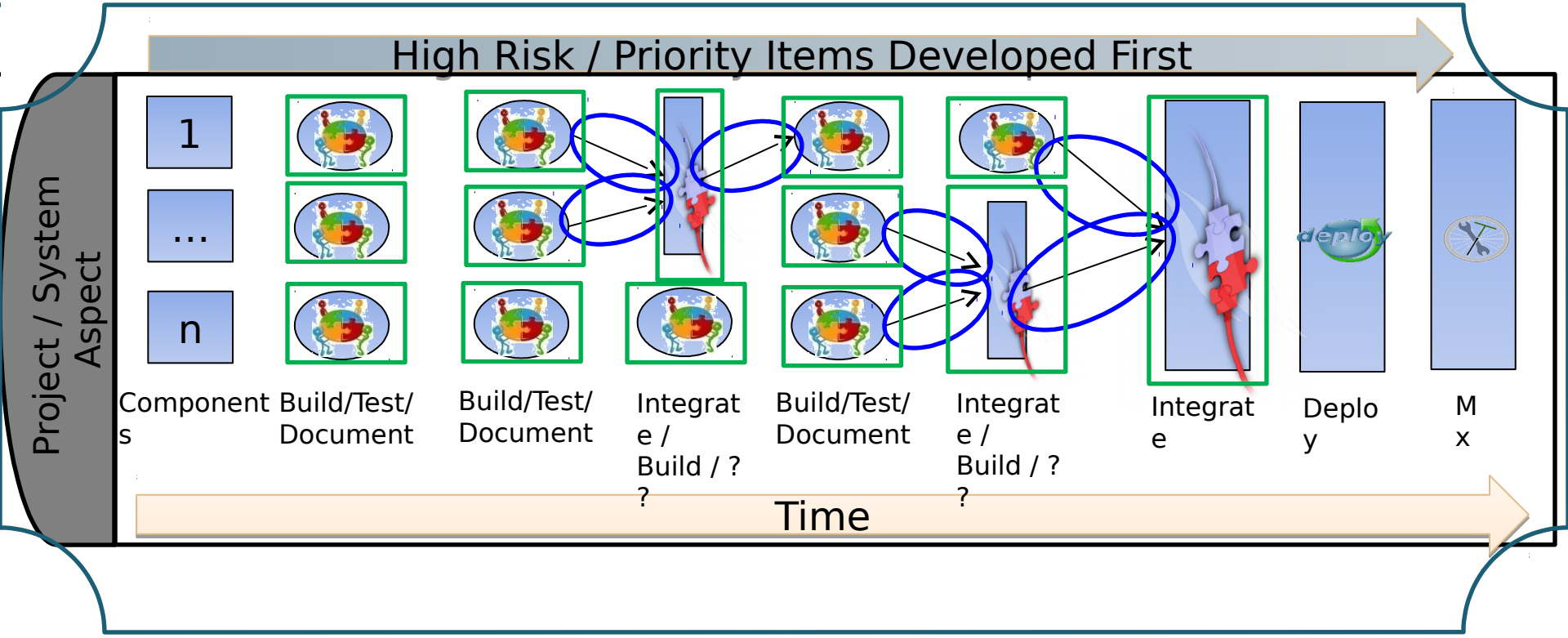
Incorporates System Engineering Best Practices

Project / System Aspect

Configuration Mgmt.

Interface Mgmt.

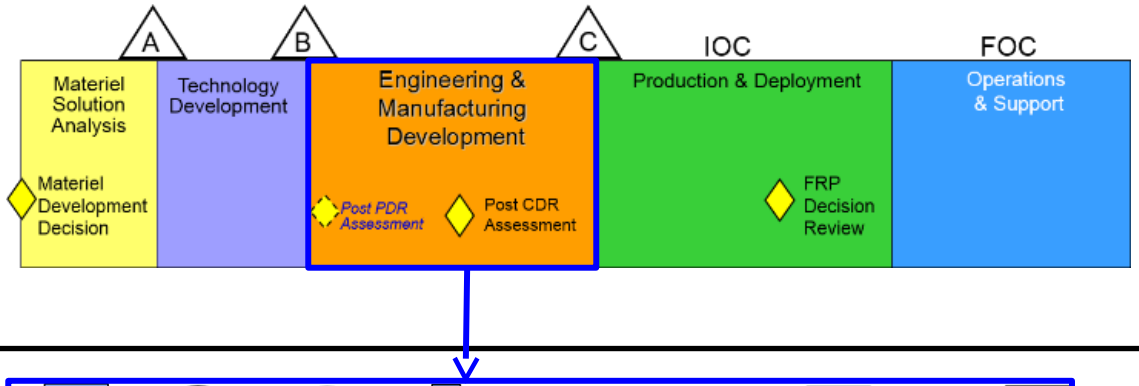
Risk Mgmt.



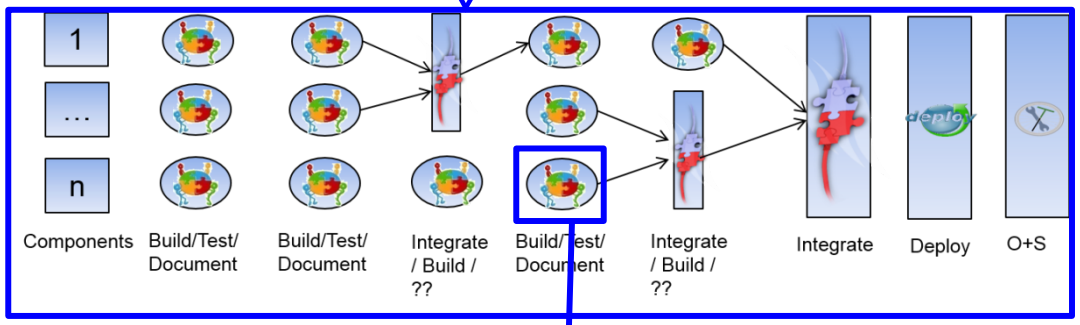
Holistic View

Project / System Aspect

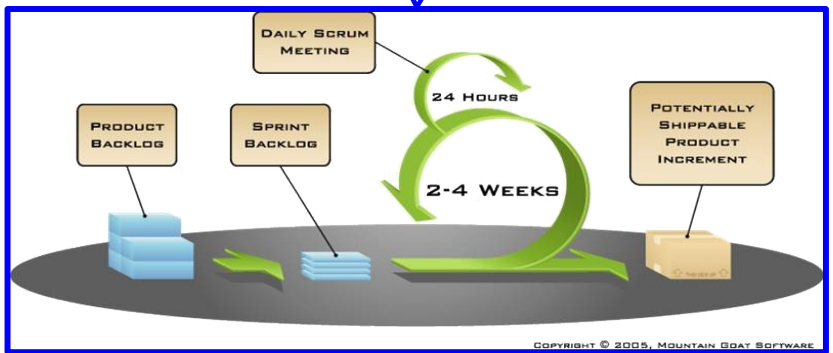
Business Aspect



Project / System Aspect



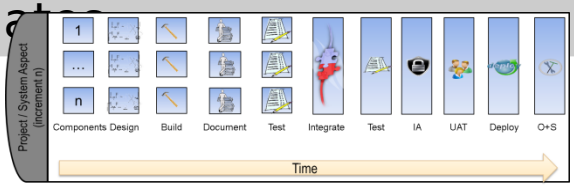
Development Aspect



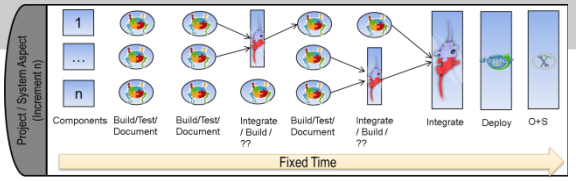
Traditional vs. Agile Project Management

Project / System Aspect

Traditional	Agile
Issues discovered later in development lifecycle	Issues discovered earlier in development
Single Integration	Periodic / Continual Integration
Structured by Domain	Multidisciplinary Teams
Adds Time / Money to release to adapt to issues / requirements	Reduces functionality in release (pushed to another release) to adapt to issues / requirements
Tracks Progress by Stages	Tracks progress by Value (functionality complete)
‘Success’ is measured by conformance to a schedule	‘Success’ is measured by delivering useful capabilities
Project Management Provides Estimates	Team Provides Estimates

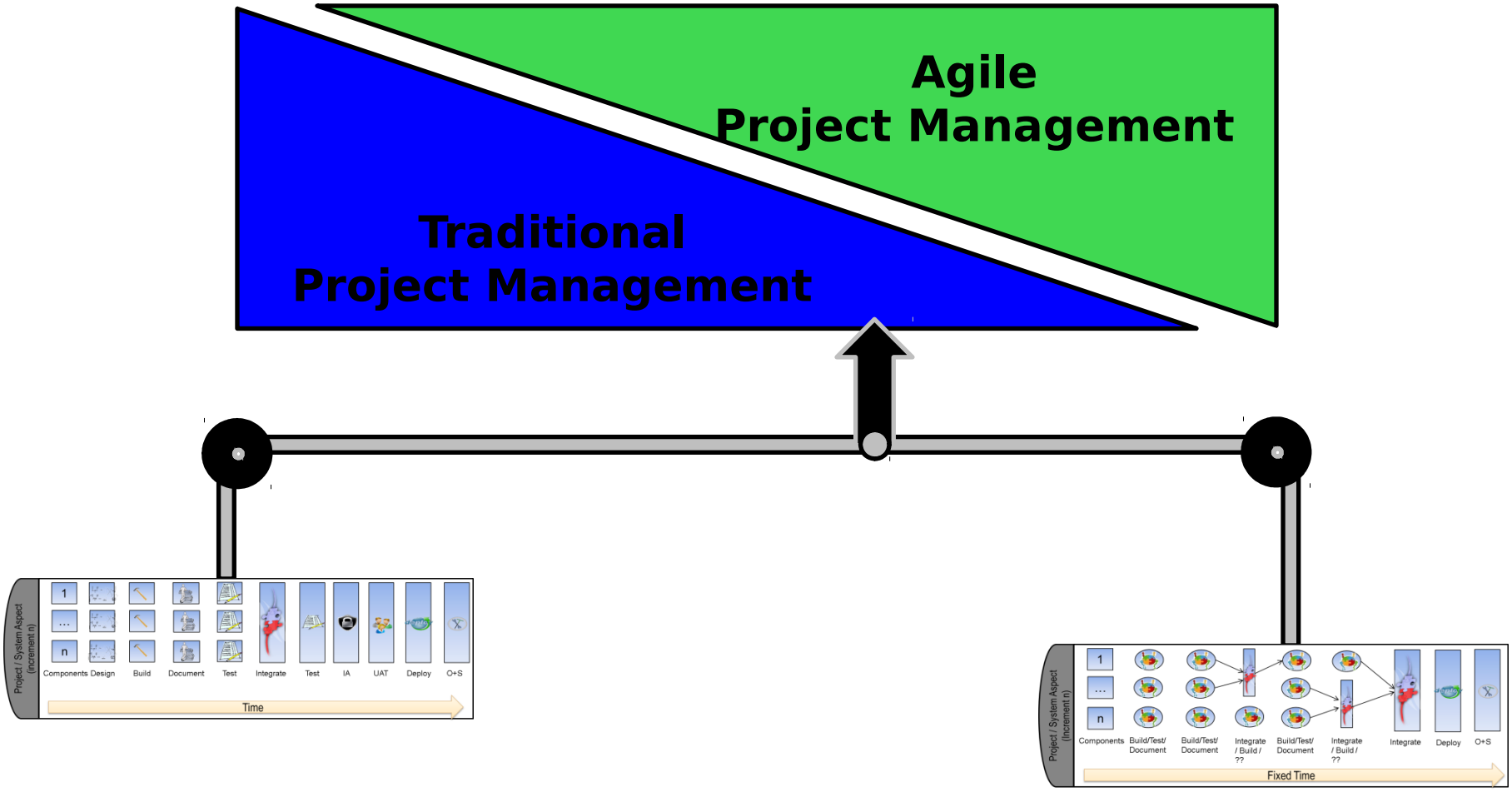


vs



The PM Spectrum

Project / System Aspect



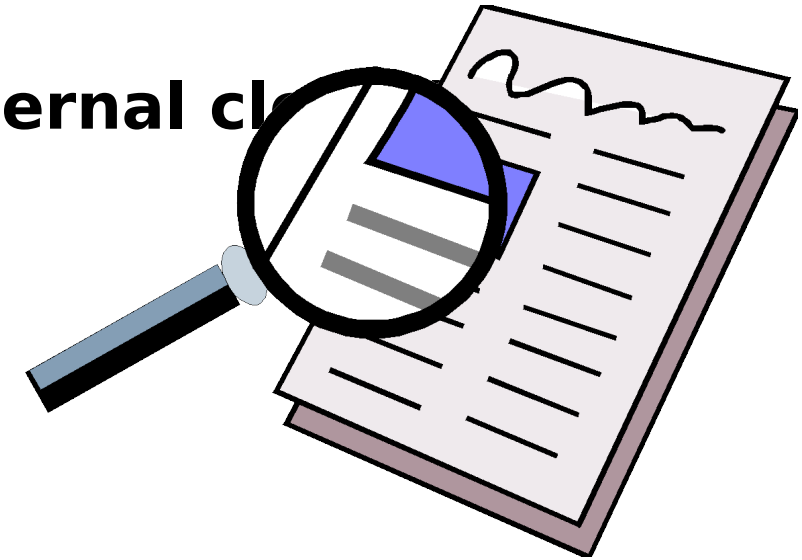
Modified from:

Example Project Management

Learn. Perform. Succeed.

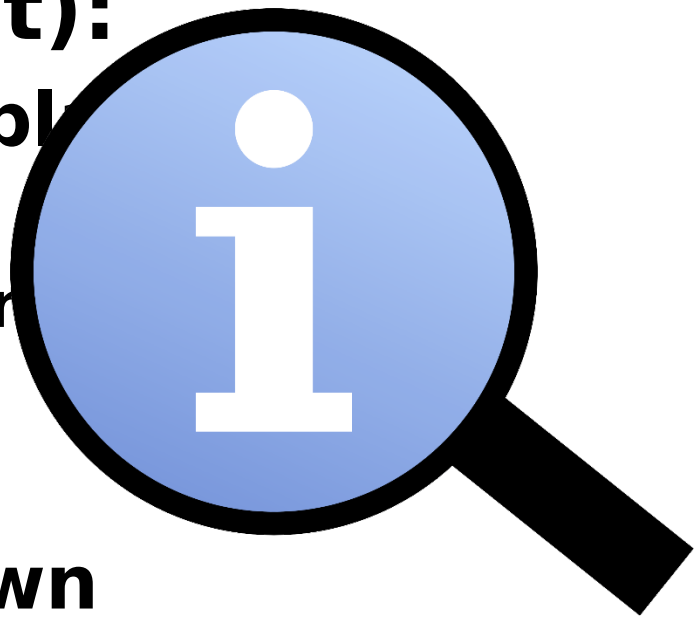
Agile Project Management Example

- **Project Description**
  - **IT Health system / software integration and development effort**
    - **10 Contracts with different start / end dates**
      - **9 T&M**
      - **1 Fixed Price**
    - **7 Systems (one external cl**



Agile Project Management Example

- **What we had (input):**
  - **The teams were in pl**
    - **Development**
    - **Program Managemen**
    - **Architecture**
    - **Systems**
  - **Contract types known**
  - **Operational Requirements Defined**



Project Team Composition

Core Project Team*

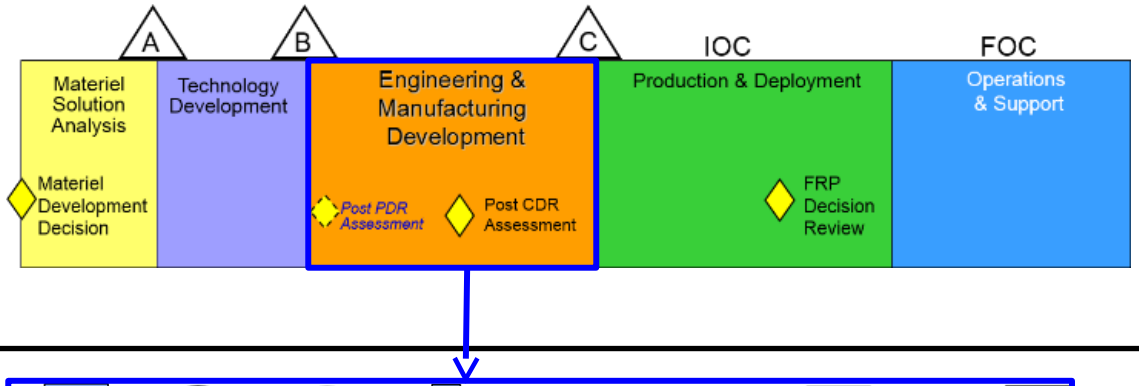
- Project Manager
- Systems Engineer
- Software Engineer
- Security Engineer
- Systems Architect

* Many other specialties dynamically incorporated throughout project

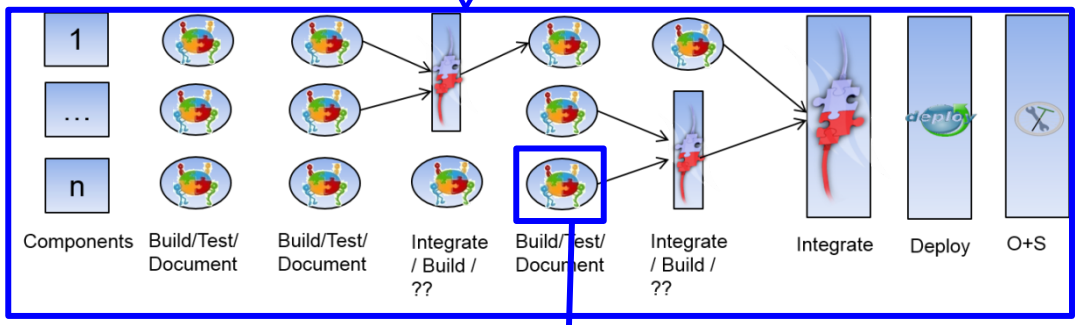
Architecture Architecture Architecture

Remember this…

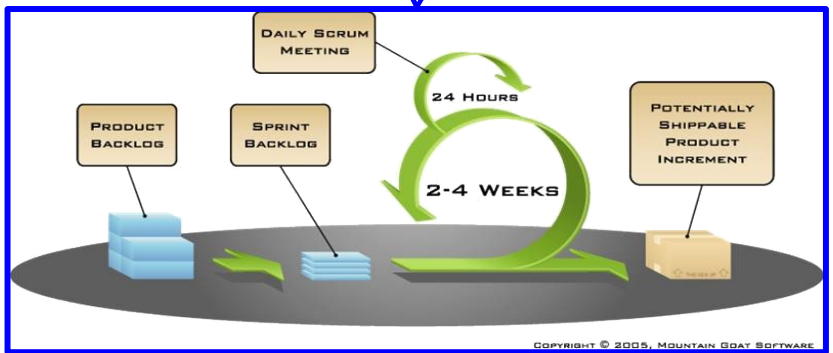
Business Aspect



Project / System Aspect

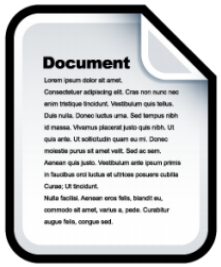


Development Aspect



Let's Use It!

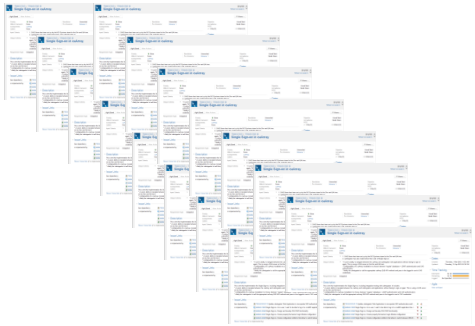
Business Aspect



Requirements
Constraints: Contracts, Funding, etc…

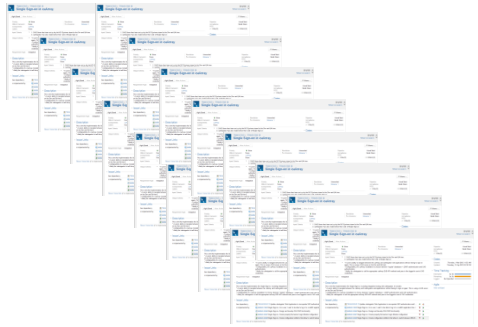
Project / System Aspect

Core PM Team



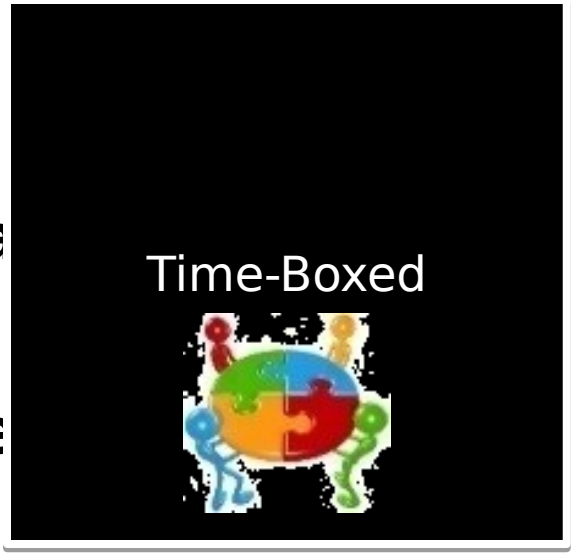
Work Packages

Developm ent Aspect

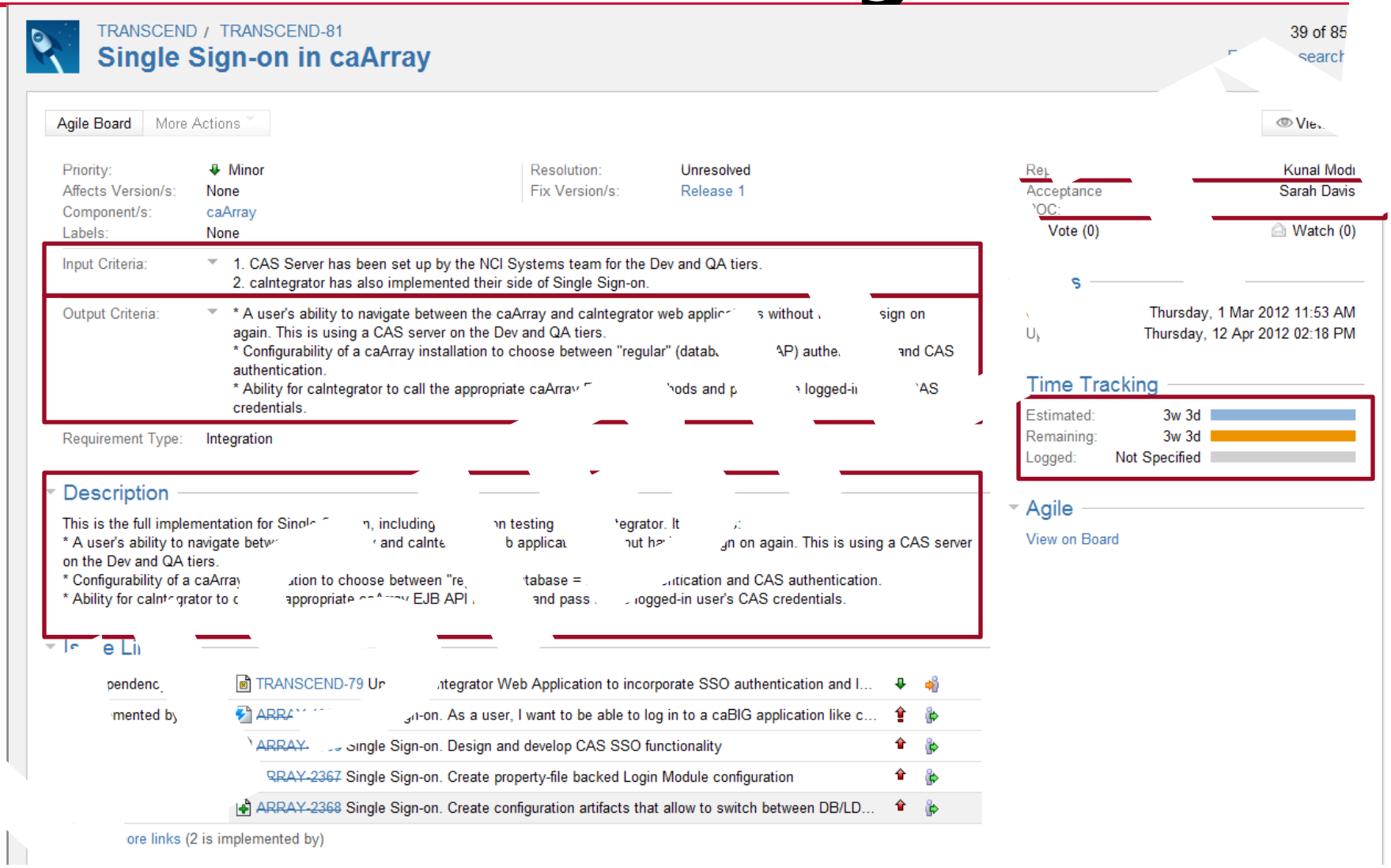


Managing Risk
"Black-box Trust"

- **Each Work Package Needs**
  - **Input Criteria**
    - **What does the team need to perform the work?**
  - **User(s)**
    - **Who will be validating the work once complete?**
  - **Time-boxed**
    - **How long will the work ta**
  - **Exit Criteria**
    - **What should be complete of the Sprint?**



Integration Work Package



TRANSCEND / TRANSCEND-81

Single Sign-on in caArray

39 of 85

Agile Board | More Actions

Priority: Minor
Affects Version/s: None
Component/s: caArray
Labels: None

Resolution: Unresolved
Fix Version/s: Release 1

Kunal Modi
Sarah Davis
Vote (0)
Watch (0)

Input Criteria:
1. CAS Server has been set up by the NCI Systems team for the Dev and QA tiers.
2. caIntegrator has also implemented their side of Single Sign-on.

Output Criteria:
* A user's ability to navigate between the caArray and caIntegrator web applic... without ... sign on again. This is using a CAS server on the Dev and QA tiers.
* Configurability of a caArray installation to choose between "regular" (datab... AP) authe... and CAS authentication.
* Ability for caIntegrator to call the appropriate caArray ... ods and p... logged-i... AS credentials.

Requirement Type: Integration

Thursday, 1 Mar 2012 11:53 AM
Thursday, 12 Apr 2012 02:18 PM

Time Tracking

Estimated: 3w 3d
Remaining: 3w 3d
Logged: Not Specified

Description

This is the full implementation for Singl... n, including ... n testing ... egrator. It ...:
* A user's ability to navigate betw... and calnte... b applica... ut ha... gn on again. This is using a CAS server on the Dev and QA tiers.
* Configurability of a caArray ... tion to choose between "re... tabase = ... ntication and CAS authentication.
* Ability for caInt... grator to c... appropriate ... EJB API ... and pass ... logged-in user's CAS credentials.

Agile

View on Board

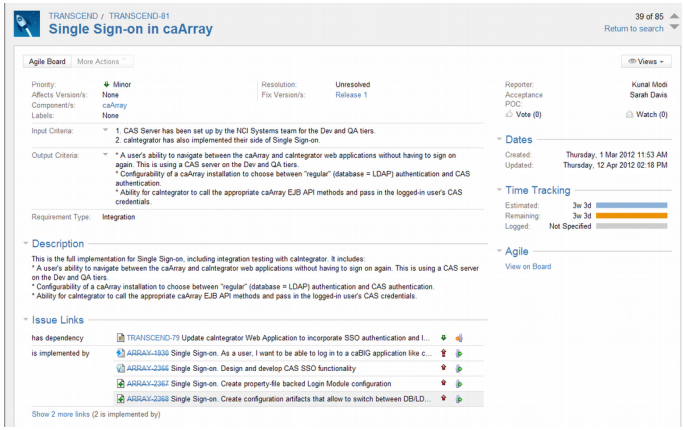
Issue Li...

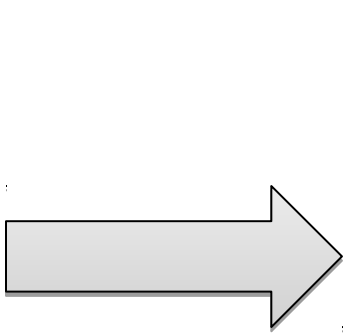
pendenc...
mented by

TRANSCEND-79 Ur... ntegrator Web Application to incorporate SSO authentication and l...
ARRA... gn-on. As a user, I want to be able to log in to a caBIG application like c...
ARRAY-... Single Sign-on. Design and develop CAS SSO functionality
ARRAY-2367 Single Sign-on. Create property-file backed Login Module configuration
ARRAY-2368 Single Sign-on. Create configuration artifacts that allow to switch between DB/LD...

ore links (2 is implemented by)

Work Package(s) were Assigned to Team(s)

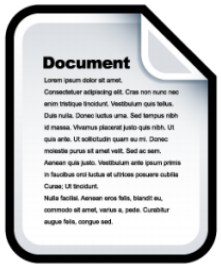

TRANSCEND / TRANSCEND-81
Single Sign-on in caArray
39 of 85
Return to search
Agile Board
More Actions
Views
Priority: Minor
Affects Version/s: None
Component/s: caArray
Labels: None
Resolution: Unresolved
Fix Version/s: Release 1
Reporter: Kunal Modi
Acceptance POC: Sarah Davis
Vote (0)
Watch (0)
Input Criteria:
1. CAS Server has been set up by the NCI Systems team for the Dev and QA tiers.
2. calntegrator has also implemented their side of Single Sign-on.
Output Criteria:
* A user's ability to navigate between the caArray and calntegrator web applications without having to sign on again. This is using a CAS server on the Dev and QA tiers.
* Configurability of a caArray installation to choose between "regular" (database = LDAP) authentication and CAS authentication.
* Ability for calntegrator to call the appropriate caArray EJB API methods and pass in the logged-in user's CAS credentials.
Requirement Type: Integration
Dates
Created: Thursday, 1 Mar 2012 11:53 AM
Updated: Thursday, 12 Apr 2012 02:18 PM
Time Tracking
Estimated: 3w 3d
Remaining: 3w 3d
Logged: Not Specified
Agile
View on Board
Description
This is the full implementation for Single Sign-on, including integration testing with calntegrator. It includes:
* A user's ability to navigate between the caArray and calntegrator web applications without having to sign on again. This is using a CAS server on the Dev and QA tiers.
* Configurability of a caArray installation to choose between "regular" (database = LDAP) authentication and CAS authentication.
* Ability for calntegrator to call the appropriate caArray EJB API methods and pass in the logged-in user's CAS credentials.
Issue Links
has dependency
TRANSCEND-79 Update calntegrator Web Application to incorporate SSO authentication and l...
is implemented by
ARRAY-1930 Single Sign-on. As a user, I want to be able to log in to a caBIG application like c...
ARRAY-2366 Single Sign-on. Design and develop CAS SSO functionality
ARRAY-2367 Single Sign-on. Create property-file backed Login Module configuration
ARRAY-2368 Single Sign-on. Create configuration artifacts that allow to switch between DB/LD...
Show 2 more links (2 is implemented by)



Capability Focused

Project Planning

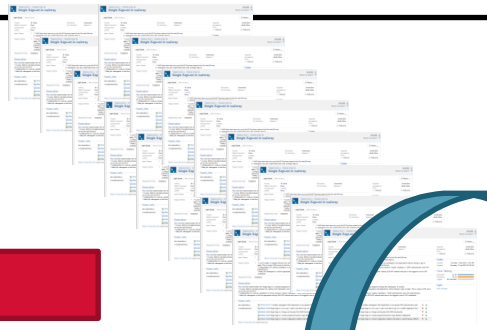
Business Aspect



Requirements
Constraints: Contracts, Funding, etc…

Project / System Aspect

Project Estimate



Work Packages

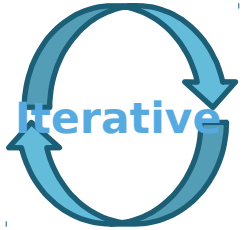
Iterative

Software Aspect

Feedback and Time Estimates for each work package

Hours / Days NOT Weeks

Iterations of Work Packages

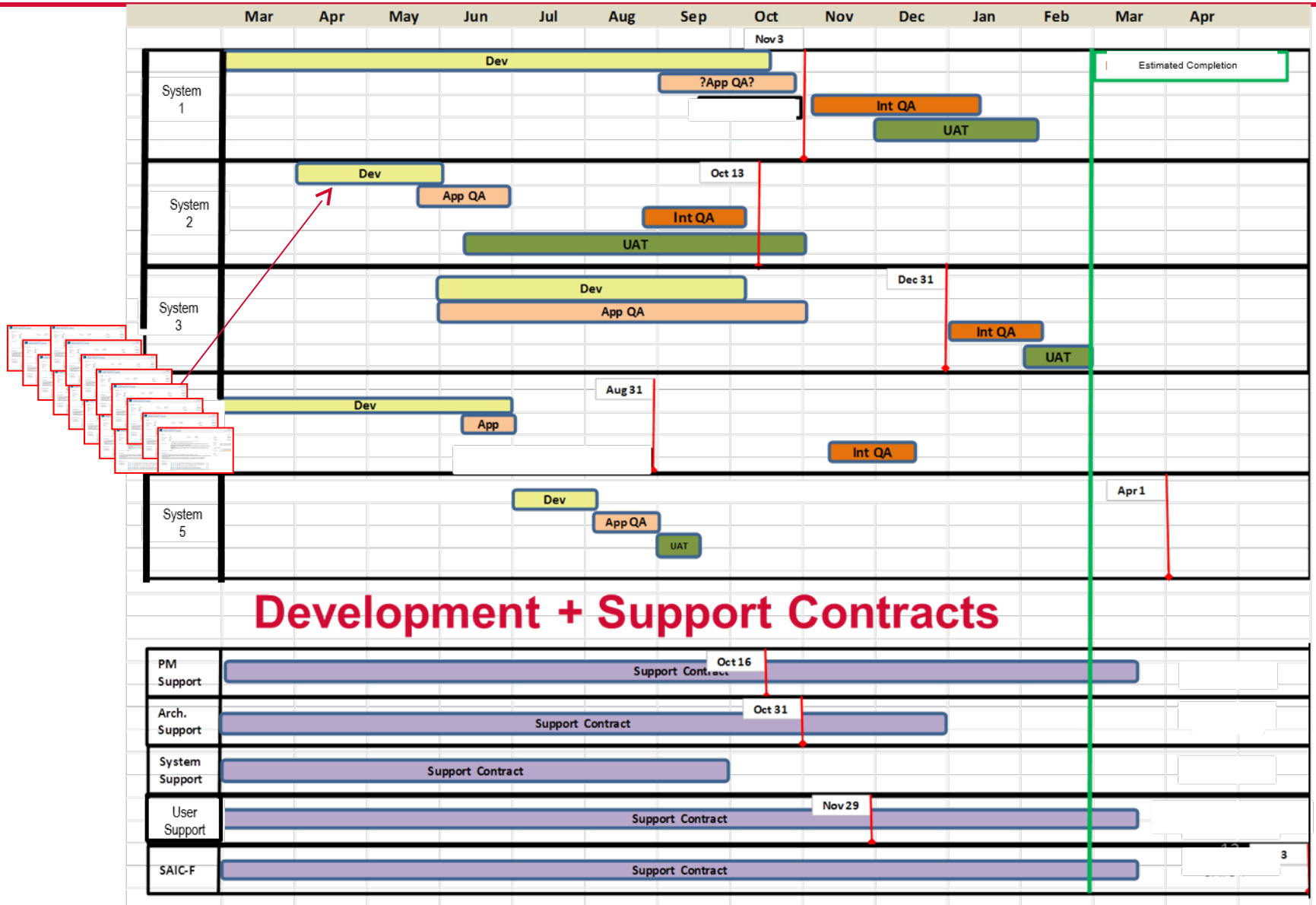


- All of the work packages contained enough information for that work to be completed - some contained architecture diagrams and / or standard specifications
- Review / Negotiation of time estimates
  - Either too high or too low raises a flag
- Caused more work packages to

Things to consider while prioritizing Work Packages

- Risk: Development, contract, funding, etc…
- User needs (as defined by the user)
- Other Sprint / System Dependencies
- Contract End Dates
- Funding type / expiration
- Specialized resource availability
  - Ex., SME (Domain specific integration)

It does not look Agile?



Issues Encountered

Making The Move Toward Agile

Learn. Perform. Succeed.

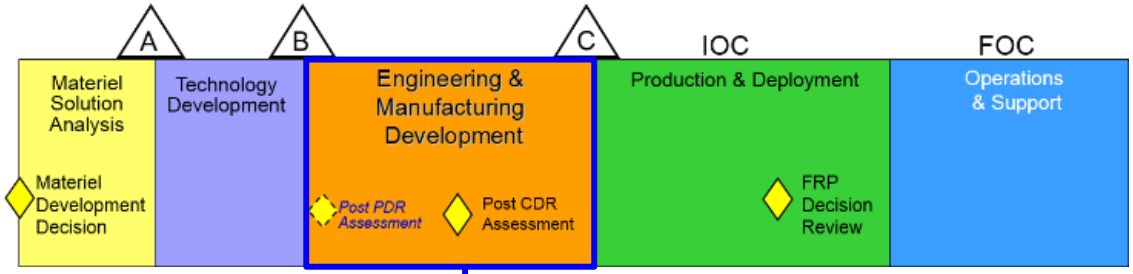
The Way Forward

1) **Define the problem to be solved**
   - **User satisfaction, quality, supportability, etc…**
2) **Figure out where you are now**
   - **Define metrics based on the problem**
3) **Identify potential solutions**
   - **Maybe agile techniques… Maybe not**
4) **Develop a plan to move forward**
5) **Execute, assess and readjust**

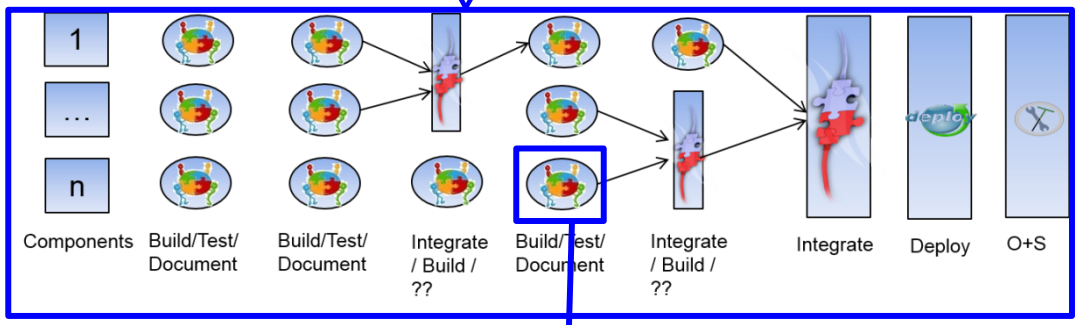
"A problem well stated is a problem half-solved." - Charles Kettering

General Training is Required at ALL Levels!

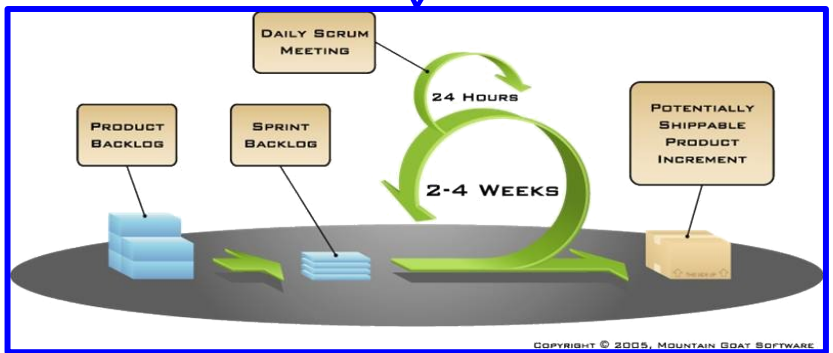
Business Aspect



Project / System Aspect

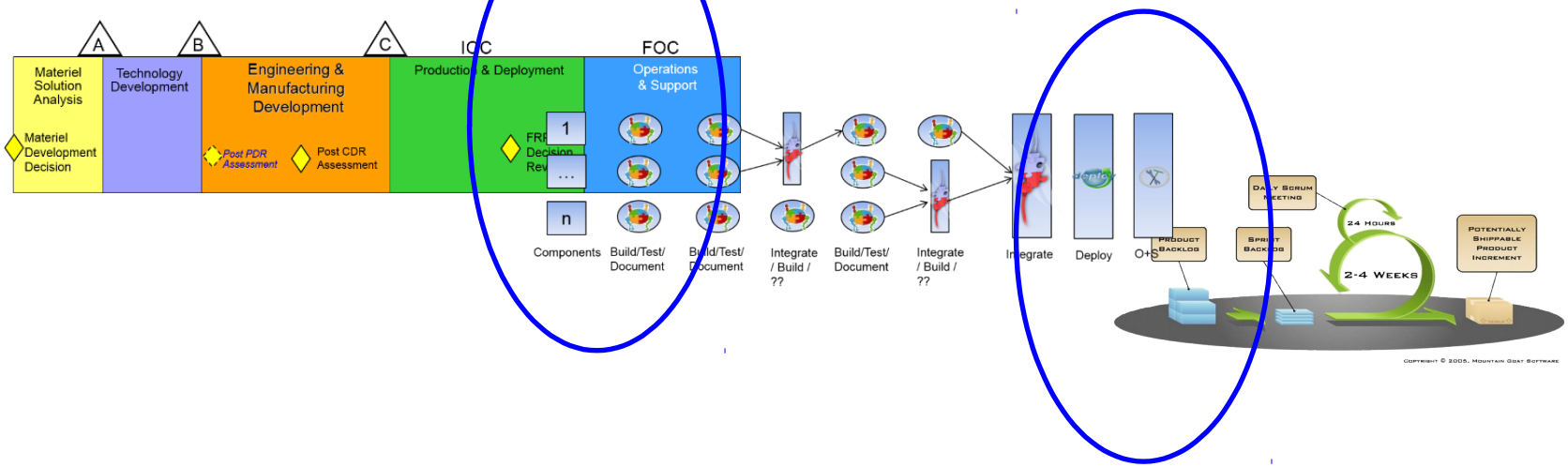


Software Aspect



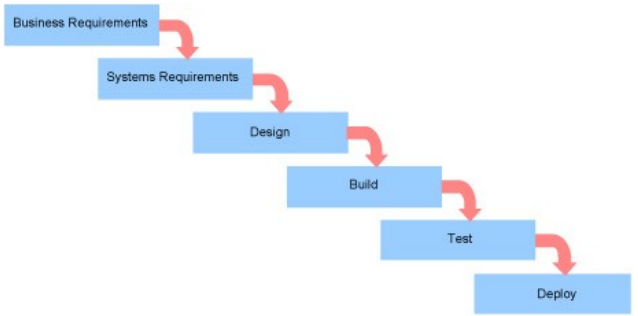
Organizational Coaching is Likely Required

- "Canned" training can only take you so far
- Your organization needs to define their path forward
  - Metrics, reporting, etc…



Parting Thought

- **You can’t use agile project management without an agile project!**



Matthew R. Kennedy, PhD, CSP
Matthew.Kennedy@dau.mil